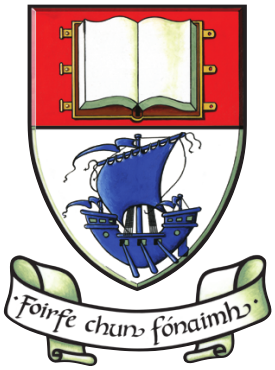
Distributed Learning and Data Processing for Smart Farming



Dixon Vimalajeewa (No. W20073183)

Department of Computing and Mathematics

Waterford Institute of Technology

Thesis submitted in partial fulfilment of the requirements for the award of

Doctor of Philosophy

Supervisors: Dr. Sasitharan Balasubramaniam and Dr. Donagh P. Berry

Submitted to Waterford Institute of Technology, December 2019

I would like to dedicate this thesis to my family who have been my inspiration and strength throughout the course of this study.

Declaration

I hereby declare that this material, which I now submit for assessment on the programme of study leading to the award of Doctor of Philosophy, is entirely my own work and has not been taken from the work of others save to the extent that such work has been cited and acknowledged within the text of my work.

Dixon Vimalajeewa
Submitted to Waterford Institute of Technology, December 2019

Acknowledgements

I would like to acknowledge my supervisors Dr. Sasitharan Balasubramaniam and Dr. Donagh P. Berry for their guidance, support, encouragement, and patience throughout my studies. I will be forever grateful for all your help and motivation that made my PhD experience productive and stimulating. I am extremely thankful for the excellent example that you have provided as great researchers. Also, I am thankful to Dr. Chamil Kulatunga for being my supervisor in my first year and for his continuous support and guidance for completing my studies. All the while, I was instilled with the comfort of knowing that you and your experience were always at hand during times of difficulty.

I would like to express my gratitude to my former research unit manager at TSSG, Mr. Eric Robson, for his enthusiastic support and encouragement, and the entire Artificial Intelligence and Machine Learning Group, especially, the Precision Dairy team - Dr. Deirdre Kilbane, Dr. Stepan Ivonov, and Dr. Kriti Bhargava for their immeasurable help, motivation, and advice over the years. I extend my gratitude to the entire team and staff at the TSSG, and the wider community at WIT, especially my mentor Dr. Bernard Butler for his continuous assistance during my time as a student. The funding of Precision Dairy (ID: 13/1A/1977) and VistaMilk (ID: 16/RC/3835) provided by SFI is also gratefully acknowledged.

I thank my parents and my siblings for being a constant source of moral and emotional support. Your encouragement, love and faith gave me the strength to pursue my dreams. Last but not the least, I thank all my friends from Sri Lanka, Finland and Ireland for their constant motivation, help and wonderful memories over the years.

Abstract

Smart farming (SF) is a sustainable farm management concept used for the intensification of food production to meet the growing demand. With the progress of the Internet of Things, advanced systems have been widely proposed for monitoring and processing data to generate insights that help producers to optimize farm management processes. Centralizing data to a remote Cloud is the conventional data processing method, though the extended latency in getting insights back to the application and intermittent Internet connectivity limit its adoption particularly in time-sensitive applications. Alternatively, distributed data analytics methods have been introduced to enable processing data in proximity to the sources and then combine insights accordingly for making timely and accurate decisions cooperatively. However, most of the SF systems currently in use operate in isolation due mainly to the lack of analytic techniques that can effectively incorporate them for processing data. Consequently, their full potential as well as the data collected by them is significantly under-utilized.

This PhD research focuses on the development of distributed data processing and learning methods to enable cooperative data analytics. Initially, this research explores how large-scale complex data can be simplified for conducting effective analysis and then proposes a Compressed Learning (CL) approach and a novel metric, known as animal importance (AIm), to extract meaningful information to perform learning effectively. To illustrate the potential of the CL approach in processing large-scale data in the SF domain, this study presents an application of CL in analyzing large-scale Mid-Infrared (MIR) milk quality data. Also, as an application of the AIm metric in the smart dairy farming domain, the research discuses how effectively AIm could be used for alerting the prevalence of sick and estrus cows in a herd based on the variability in behavioral dynamics. Second, this PhD research develops a hybrid model to mitigate drawbacks that limit using conventional machine learning models and proposes the Federated Learning (FL) method to train distributed data sources cooperatively. The FL-based system is analyzed to determine its applicability for assessing milk quality by incorporating MIR milk quality data collected at distributed farms. This is then followed by considering the fact the limitations of the FL-based approach when it comes to making the data analytics more trustable and transparent to every participant in the

distributed network, by integrating a Block Chain-enabled fully decentralized distributed learning framework. In particular, this framework integrates the Internet of Nano Things (IoNT) that has previously not taken into account any Block Chain-enabled system. The proposed framework is then used for monitoring the level of chemicals (e.g., fertilizers) on farmlands. Finally, this PhD research discusses optimum utilization of available resources in cooperative distributed data analytics by offloading computations to neighboring devices. Computation offloading enhances the timeliness and learning accuracy in cooperative data analytics as well as enabling the efficient use of limited energy resources found in sensor devices, and this includes solar energy harvesting devices.

.

Contents

Chapter 1

Introduction

1.1 Background and Motivation

This section provides the background and motivation for the research work presented in this thesis. First, section 1.1.1 provides a general overview of the agriculture sector (agri-sector) along with its current challenges. The thesis builds on these challenges by discussing solutions that have been proposed for Smart Farming in section 1.1.2, and this is followed by a discussion on how data analytics can contribute towards addressing these challenges in section 1.1.3.

1.1.1 Agri-Food Production and Challenges

Agriculture is the oldest manufacturing sector and still the major livelihood for nearly 40% of the world population [1]. It also plays a vital role in feeding the world, while making a significant impact on society as well as the economy. In the past, mechanization of the agri-sector contributed to a considerable shift in efficiency, productivity and sustainability. The study in [2], for instance, reported that the world crop yield increased by 70% and the greenhouse gas emission dropped by 23.8% during the period 1960-2007 and 1990-2012, respectively. While these advancements have primarily focused on increasing food productivity, it is not sufficient in mitigating the critical challenges raised in parallel to the rapid expansion of the agri-sector. A number of these critical challenges include:

- **Population growth:** Based on the statistics provided by the Food and Agriculture Organization (FAO) of the United Nations, the world population is expected to be around 9.2 billion by 2050, and consequently, the food demand will be increased by

70% [3]. At the same time factors such as growing urbanization and increasing income are accelerating the world food demand.

- **Food loss and waste:** Nearly one-third of food produced for human consumption is wasted annually. This causes economic losses of around 940 billion dollars, along with wasting natural resources and increasing the CO_2 footprint of food products [1].
- **Environmental impact:** Resource incentive and high-input farming strategies have increased deforestation, degradation of natural resources, increased emission of greenhouse gases, and soil depletion, resulting in instability of the ecosystem (e.g., loss of biodiversity) and climate change.
- **Food quality and security:** With the growing demand for food, the use of low quality and unsafe food production practices (e.g., excessive use of chemicals) has been increased, creating critical challenges in food safety and human health.
- **Transparency in food supply chain:** Lack of collaborative and coordinated food production and distribution strategies limits the available information for effective management of food supply and demand. Consequently, this increases food waste, health issues and economic loss.
- **Competition for resources and market:** Natural resources available for farming are becoming extremely limited, though their usage is rapidly increasing due to extensive farming. The rapid expansion of the agri-sector has also created market competition which negatively impacts the small-scale farm holders who find it difficult to compete with large-scale multinational companies.

Alternatively, the recent developments in the Internet of Things (IoT), Internet of Nano Things (IoNT) and Information and Communication Technologies (ICT) are being widely proposed to overcome these limitations. These technologies provide great opportunities to convert the agri-sector into a smart sector, by developing innovative approaches to smart food production and distribution strategies. Therefore, the agri-sector is aiming today to integrate these technologies in order to address the growing food demand through sustainable intensification of productivity. For example, it is expected that the use of IoT devices in agriculture will reach up to 75 million by 2020 [1]. The Smart Farming concept is the realization of integrating modern technology in the agri-sector in order to balance the environmental costs while aiming to feed the current and future population.

1.1.2 Smart Farming (SF)

Smart Farming (SF) is a farm management concept that is used to employ sustainable farm management that aims to create greater production and profit with minimum environmental impact and waste. In doing so, modern IoT devices such as sensors, mobile phones, robots, and various ICT infrastructures will collectively monitor the spatial and temporal variability (e.g., grass growth, soil fertility, weather and animal well-being) during the farming process. The collected data is then processed and converted into actionable insights that help producers to make meaningful and timely informed decisions to optimize the farm management process. Today, SF is not only about intensifying farm productivity but also maintaining a transparent and credible food supply chain. Hence, farm management as well as supply chain processes such as consumer buying behavior and satisfaction and variability in demand for different food items are carefully monitored and managed as a single system. This means that the collected data is integrated and processed cooperatively to derive insights that can empower the collaborative operation of the entire food production and supply chain system. An example of this is adjusting dairy milk production based on the market demand as well as prevailing weather conditions, while maintaining optimum animal well-being (see Figure 1.1). This new form of connected system can bring several benefits to the agri-sector, and examples of this are listed as follows:

- **Community farming:** Producers can work together as a community to improve the quality and effectiveness of their farming practices. This is particularly beneficial for rural farming where there are limited resources. For example, IoT technologies can be used to produce a common data storage to share information, thereby increasing the interactivity between farmers, retailers, and agri-experts.

- **Operational efficiency:** Continuous monitoring of SF processes can enable early warning of health issues and adjustments in the production and supply chain depending on the variability in the market demand.

- **Cost and wastage:** Remote monitoring and self-adaptability of a large number of IoT devices deployed in SF can contribute to reducing human labor and saving money and time required for farm management. Such autonomous functionality allows for on-demand supply of farm inputs such as fertilizers, feed, and water. This helps to reduce farm input wastage significantly.

- **Awareness:** Collaborative SF enables everyone (e.g., farmers, consumers) to be aware of the whole food production and supply chain practices. For example, producers are

well aware of current food prices and demand, market competition, overall environmental impact. Consumers can also have opportunities to access information such as food quality, safety and freshness required in order to determine the quality of products that they are going to buy.

Therefore, SF has a great potential in mitigating the challenges mentioned above and addressing the world's increasing food demand.

Various advanced systems are available today in the agri-sector to provide intelligent services to enable autonomous and remote control of SF operations in order to improve the SF practices. These systems primarily consist of three main components: IoT devices, supporting communication infrastructure, and data storage and processing facilities. Integrating these elements creates a smart web of connected objects (e.g., wireless sensor network (WSN)), enabling monitoring of various farming processes that can possibly be controlled remotely. The IoT devices are mostly sensors which are mainly used for monitoring different parameters such as weather, soil nutrients and food quality. Communication technology plays a vital role in deploying IoT systems, and some of the commonly used communication technologies are low power wide area (LPWA), Bluetooth and WiFi. The Internet paves the way to exchange data over the network and enables data to be available anywhere and anytime. Cloud, Fog and Edge computing methods coupled with advanced data analytic methods such as Machine Learning (ML) and Artificial Intelligence (AI) are used to process data in order to derive insights for decision-making. For instance, *DairyMgt*, a suite of decision support systems (DSSs) developed for smart dairy farming sectors, consists of decision support tools that enable monitoring of animal nutrition and feeding, production and productivity, environmental stewardship, price risk management and financial analysis [4].

The progressive involvement of such advanced systems accelerates the datafication of the agri-sector. Consequently, the whole agri-food production and supply chain have shifted towards data-driven and data-enabled frameworks which has propelled new innovations in disruptive technologies. For example, the use of emerging nanotechnology in SF applications for monitoring nutrients in soil has expanded the spectrum of data from the macro-scale to the nano-scale, creating opportunities to gain novel insights and deeper knowledge [5]. Thus, the knowledge acquired from data plays a vital role in empowering sustainable farm productivity and supply chain operations to meet increase in food demand. Therefore, analysis of SF data would not only help to improve the effectiveness of farm practices but will also stimulate the innovation of novel strategies to employ more knowledge-based food production and supply chain operations.

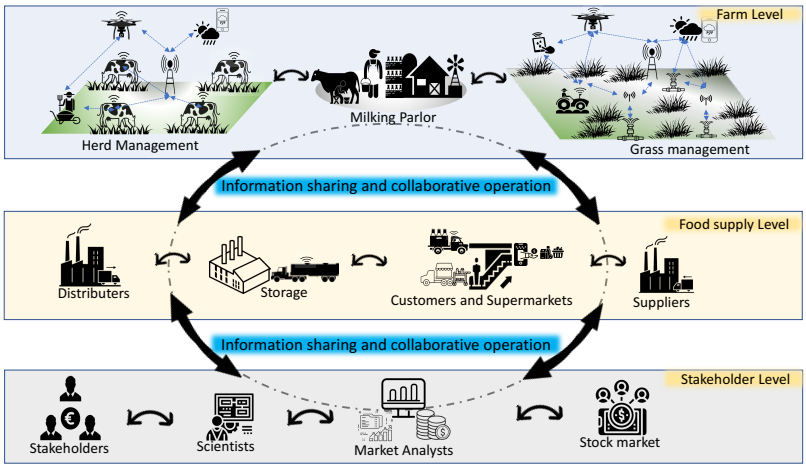


Fig. 1.1 Data analytics at different levels of the food production and supply chain.

1.1.3 Data Analytics in Smart Farming

The greater prevalence of modern IoT devices, hardware and software systems, and supporting communication technologies can enable generation of data from diverse sources at different levels in the SF process. This may include data sources from animals/plants-, the farm environment-, production facilities-, and end-product-levels (Figure 1.1). Different communication technologies described in section 1.1.2 can be used for efficient delivery of data to infrastructure storage (and processing) resources such as remote Cloud and data centers. Consequently, large amounts of heterogeneous and complex datasets are collected in three different forms; structured, semi-structured, and unstructured.

The data analytic process may vary depending on the complexity and nature of the data to be analyzed. At the same time, data analytic process is executed to suit application-specific requirements such as resource availability on the device. Therefore, these two factors are inter-related and various ML techniques and models are available for effective analysis of large-scale complex data under diverse application specific requirements.

1.1.3.1 ML Techniques

Considering the complexity and nature of data, modern data science uses several advanced ML techniques. Example of these techniques include Representative learning, deep learning, transfer learning, distributed and parallel learning, and Active learning [6]. Representative learning can effectively deal with higher dimensional data as it enables deriving a common representation by using different data analysis techniques such as classification and feature selection. Transfer learning is designed to learn from datasets that have different distributions, while distributed and parallel learning can deal with multidisciplinary datasets. Deep learning

is a promising method for extracting features from both structured and unstructured data. However, it requires massive amounts of data for achieving higher accuracy and collecting such large datasets has proven to be difficult in some applications. As an alternative, active learning is commonly used as it can achieve higher accuracy with limited data.

As far as application requirements are concerned, different learning approaches are used to derive insights, and primarily, they are divided into four categories; real-time data analytics, off-line data analytics, memory-level data analytics, and massive data analytics [6]. Real-time data analytics is used when immediate responses are required. This can include time-sensitive (streaming) data that has to be processed in real-time based on the application deadline in order to generate and distribute insights. For example, the distributed and parallel learning method can be used to adjust irrigation systems according to daily weather changes. In certain cases, long-term data collection is required in order to take action to optimize farming operations in response to climate change. In such cases, off-line data analytics is used as there is no need for immediate responses (e.g., historical data analysis), and representative learning is a good fit. When there is sufficient memory space to perform data analytics compared to the data volume, memory-level data analytics is used. In situations where there is a large data set, massive analytic approaches such as Cloud computing are commonly used and deep learning is one of the most famous models utilized.

1.1.3.2 ML Tasks

The ML techniques mentioned above use different tasks to extract insights based on different models and they are mainly identified as classification, clustering, association rule mining, and predictions [3]. These ML tasks are further categorized as supervised, semi-supervised, and unsupervised depending on whether the data used for the task is labeled or not. Classification is a supervised learning technique widely used for identifying groups that represent mutually distinct characteristics. For instance, identifying groups of animals which may have nutrient deficiencies will require classification. The most commonly used ML models for classification are Artificial Neural Network (ANN), Support Vector Machine (SVM), k-nearest neighbor (kNN), and Bayesian models [7]. Clustering is similar to classification, but it is an unsupervised approach and mostly used for grouping massive datasets into smaller subgroups which can easily be used for further analysis. Identifying inter-relationships among different parameters is vital for generating robust insights and in doing so, the associate rule mining is a promising method. Prediction is the most commonly used supervised learning method to identify, for instance, food consumption trends. Time series based ARMA and ARIMA models, regression, ANN, deep learning, and fuzzy logic techniques are some of the

commonly used ML models for prediction. In SF, these methods can be used to warn farmers about the prevalence of diseases and extreme weather conditions and customer buying trends.

Availability of sufficient resources is one of the most critical factors that need careful attention before using these data analytics techniques. Based on both the nature of the data and application requirements (e.g., timeliness of learning outcomes) and availability of computing, communication and storage resources, these learning techniques are mainly used under two schemes: centralized and distributed ML (CML and DML). With the advances in modern ICT and IoT devices and data mining methods, the development of novel CML and DML based analytic platforms have gained phenomenal success in recent years.

Centralized Machine Learning (CML)

CML is the most dominant ML approach and facilitates the extraction of meaningful insights from complex data in more precise and effective manner. Most sensor technologies and IoT platforms provide services to collate and store vast quantities of data collected from geographically distributed sources. Computational facilities for analyzing such data mostly reside in centralized data centers (e.g., Cloud) where data will be consolidated as single large datasets and analytics can subsequently be performed. It is widely accepted that CML is highly advantageous for developing new hypotheses, as it enables improved learning accuracy and model acceptability [8]. The most widely used CML approach is Cloud computing [9]. It facilitates on-demand access to computational and storage requirements in remote servers via the internet. Furthermore, the Cloud infrastructure provides computing facilities by utilizing resources based on the load of the tasks and avoids over and under-provisioning of resources. Thus, the infrastructure can save capital expenditure and in particular the energy costs.

However, the unprecedented volumes of data collected by IoT devices can overwhelm the storage, computing, and communication capabilities of existing CML-based data analytic platforms. This causes unnecessary delays in communication and computation and can affect applications in rural agriculture where there is intermittent connectivity between the Cloud infrastructure and farms. This results in extended latency, making CML platforms unsuitable for time-sensitive and real-time analytic applications. At the same time, CML suffers from other challenges such as single point failure, storage, energy consumption, data heterogeneity and security problems. As a promising alternative to mitigate these shortcomings, DML platforms have gained a considerable attention in modern-day data analytics.

Distributed Machine Learning (DML)

DML is based on the data gravity concept that uses computing, storage and networking resources in close proximity to the data sources. This means the data analysis are performed synchronously in close proximity or on the device that generated the data. This model provides new opportunities for extracting insights from large-scale data in a real-time manner with minimum need of centralized data collection, while optimally utilizing distributed computing, storage, and communication resources. Thus, DML minimizes the dependency on intermediate (or a third party) entities such as Cloud servers, which in turn facilitates preservation for data security, privacy and ownership, which are identified as the most critical challenges in modern-day data analytics applications. Therefore, DML can ensure timely and accurate decisions by collaborating with pervasive data sources. Fog- and Edge-computing are new paradigms that are utilizing DML methods and work in conjunction with Cloud computing systems [10]. They provide opportunities for partial migration of computing tasks away from the Cloud towards the edge of the network. Fog or Edge devices with deployed intelligence are capable of executing light-weight and ML tasks (e.g., aggregation, classification) and making time-sensitive decisions accordingly. In parallel to this, certain data and learning outcomes are also sent to the Cloud for long-term storage and further analysis. In doing so, they contribute to reducing application latency and minimize data communication cost for transmitting data to the Cloud.

Based on these computing paradigms, more advanced and also fully distributed DML platforms have been introduced for modern-day DML applications. For example, *Apache Hadoop* software library is a distributed data processing platform in which large datasets are processed as batches across clusters of computers. Apache also proposed *Storm* [11] and *Spark* [12] platforms for processing streaming data in order to improve the timeliness of the analytical outcomes. *MapReduce* [13] is also a widely used DML platform for parallelism of data processing tasks. Similarly, *Horovod*, a distributed learning framework, is mainly designed for *TensorFlow, Keras, PyTorch* and *MXNet* to employ distributed deep learning algorithms [14]. The most recent DML platforms, *CommentCloudCare(C*^{3}*)*, offers a suite of functionalities for conducting more transparent and privacy preserved analysis on sensitive data [15] and *Galaxy learning* framework uses Block Chain technology and Federated learning jointly for realizing fully distributed data analytics [16]. However, these computing approaches are confronted with several challenges such as as *what* and *how much* computations to offload to the edge devices in order to stabilize the trade-off between communication tasks. Further advances are, therefore, required to enhance the potential of scalability, security, autonomy and programmability of DML platforms [17].

1.2 Research Scope of the Thesis

This section discusses the scope of the PhD research presented in this thesis. Section 1.2.1 presents the challenges that current data analytic systems are confronted with and will be the main focus of the research. Section 1.2.2 will present the research objectives.

1.2.1 Challenges

Advances in data analysis techniques along with modern IoT and ICT and their applications in the agri-sector provide greater opportunities to bring more farms and stakeholders into the dynamic food production and distribution cycle (agri-ecosystem). This combination can pave the way to transforming the agri-ecosystem into a smart web (e.g., as illustrated in Figure 1.1) that can operate autonomously and controlled remotely to provide timely and intelligent services with minimal human intervention. Nonetheless, to achieve such a level of services, it is crucial to have advanced data analysis techniques that empower collaborative operations with various technologies, devices, and systems contained in the smart web (i.e., agri-ecosystem) for decision-making. However, there are several challenges where care must be taken in developing such techniques. This is because addressing these challenges will not only create opportunities for alternative solutions to mitigate problems, but it will also improve the potential to create novel opportunities for developing innovative solutions. Therefore, this section aims to explore some of the most critical challenges in the context of smart dairy farming, and they are listed as follows:

- **C1-Data Complexity:** A large number of devices continuously monitor the farming process and generate massive datasets such as milk quality data, accelerometer data of animal mobility, rumination and IR-imagery data of animal Body Condition Score (BCS). The unprecedented growth of such massive data collection exceeds the communication, computing and storage capabilities of the existing data mining platforms. Since these voluminous datasets are formed by aggregating data coming from geographically distributed data sources at different time points, they are highly complex and contain redundant and unnecessary data. Moreover, high noise accumulation, spurious correlation, and inconsistency in ML model selection are also some of the other critical issues associated with such voluminous data. These issues, however, limit the effective transferring, storing and analysis of data for extracting timely and accurate insights. More specifically, processing these data in resource-constrained environments such as in rural farming environments is highly challenging. Thus, care must be taken

for effective processing such as cleansing, dimension reduction and compression of data before applying data analytics to extract insights. It would significantly help to simplify data for conducting efficient analysis, transferring and storing. Therefore, the design of novel metrics for explaining complex data, and optimal and effective data processing methods which can optimally utilize the available resources is of utmost importance.

- **C2-Real-time Data Analytics:** SF applications are highly dynamic because the factors that affect the stability of the agri-ecosystem are highly distributed and time-variant, resulting in their significance varying over the period of SF management. This emphasizes the need for analysis of data in a real-time to capture time-variant dynamics for making timely and accurate decisions. However, the growing complexity of data limits the use of existing ML models and algorithms for conducting such data analytics as they are mostly designed for static data analytic applications under CML settings. Alternatively, many DML algorithms have been proposed, but they are confronted with critical issues such as interoperability with different ML techniques as well as distributed data sources. Thus, they have limited (or no) knowledge and flexibility in mining data by incorporating information from pervasive data sources. Consequently, these drawbacks limit the use of the full potential of the collected data as well as the available SF technologies currently in use for driving insights to employ effective SF practices. Therefore, SF requires effective and advanced ML models and algorithms that can perform real-time analytics.

- **C3-The Need to be Inter-connected:** Various devices, systems and technologies provide infrastructure to intensify SF practices and it is highly acknowledged that their collaborative operation can produce value-added services to the SF system. However, the current solutions are fragmented and poorly combined due to various reasons such as data sharing issues. Hence, they have very limited (or no) inter-communicability and operate in isolation. Consequently, the full potential of the devices and the technologies currently in use is significantly underutilized. This hinders the potential of inferring useful information by integrating distributed SF data sources. Moreover, the ML models and data analytic systems currently in use are not specially designed for SF applications, so that necessary modifications are essential to use them for a more in-depth understanding of the underlying dynamics of the SF system. However, initiatives for the development of innovative solutions for overcoming these drawbacks are still at the early stage. Therefore, the urgency of exploring solutions to allow flexible

incorporation of multiple data sources for effective decision-making is critical in the SF sector.

- **C4: Safety and Credibility of Services:** Since the SF sector is increasingly becoming a data-driven and data-enabled framework, data is one of the most valuable assets. The collaborative use of data is the key to generating precise insights required for operating efficient farm management practices. However, data owners are now more concern about the safety of their data due to the lack of data protection options deployed in data mining platforms. Limited facilities available for ensuring the transparency, integrity, validity and quality of data (and decisions) shared over the agri-ecosystem also negatively impact the validity and trustworthiness of SF services. This is because there may be malicious stakeholders who tamper, modify or delete sensitive data and share misleading information. As a result, data owners are reluctant to participate in collaborative decision-making. At the same time, valuable datasets reside in private repositories without realizing their significance in optimizing SF practices. Therefore, this emphasizes the need for novel data analytic mechanisms equipped with advanced features that guarantee the safety of data as well as the credibility of the services for the agri-ecosystem.

- **C5-Energy Efficiency:** It is widely recognized that deploying on-board intelligence in devices such as sensors used in SF improves the effectiveness of the decision-making process, such as memory-level analytics. However, such devices are mostly battery-powered, and executing analytical tasks (e.g., sensing, aggregation, and classification) is challenging as battery power is not always sufficient. Transmitting data to the Cloud infrastructure is an option, although the cost of energy for data transmission depends on the distance to the Cloud and the amount of data. Therefore, endeavors to minimize the transmission of massive datasets over long distances to the Cloud must be reconsidered to optimize energy consumption. Alternatively, efficient utilization of in-network energy is a promising solution such as offloading computations to neighboring devices based on their energy profiles. Such energy-aware cooperative computing methods have the potential to enhance the energy balance across devices and contribute to the sustainability of the SF operations. Therefore, it is of utmost importance to make efficient use of available resources.

These challenges, however, complement each other and their criticality could be varied depending on the application requirements. Care must be taken to understand these challenges and necessary actions must be taken to overcome them effectively. Challenge **C1** primarily

emphasizes the need for techniques for simplifying data complexity. The next two challenges, **C2** and **C3**, are concerned with the significance of learning algorithms that have greater potential in capturing time-variant dynamics by incorporating data generated over distributed sources. When performing such distributed analytics, the safety of data as well as the quality, validity and trustworthiness of data analytic can always be problematic due to the diversity in data collection and processing methods. Therefore, **C4** emphasizes the significance of having an analytical framework that can ensure these factors. Finally, **C5** considers the optimal utilization of energy as it is a crucial challenge, and in particular for resource-constrained SF applications.

1.2.2 Research Objectives

To address the challenges presented in section 1.2.1, this PhD research aims at developing a distributed data processing and learning framework by using state-of-the-art ML techniques, which can result in fully/semi-automated decision-making for smart dairy farming. Initially, this study explores methods for simplifying highly complex and voluminous data with minimal loss of information, thereby employing effective data mining practices. In particular, this research focuses on deriving mechanisms for efficient data compression and defining novel metrics for a more in-depth characterization of the underlying dynamics in data. Secondly, this study takes into account deriving novel ML models and then performing distributed analytics, guaranteeing data safety as well as validity of the learning outcomes. Finally, since finding sufficient resources for executing distributed data analytics is a critical challenge, the present study focuses on optimizing the available resources in a distributed computing environment. Therefore, the objectives of this research are formulated by the following Research Questions (RQ).

- **RQ1-Simplifying Data:** What are the optimal techniques and metrics for simplifying and explaining complex data with minimal loss of information?
- **RQ2-Distributed Learning:** What frameworks can be designed to make data analytics effective by securely incorporating distributed data sources?
- **RQ3-Energy Management:** How can energy be flexibly managed for systems that require distributed data analytics?

1.3 Summary

Agriculture is the one of oldest industries and the main livelihood for majority of the people. Advances in technologies has led to considerable development over the last 100 years and the agri-sector today is increasingly relying on technology. Consequently, the nature of farming and food production has changed dramatically in order to address the world's increasing food demand. In parallel to this, several challenges such as environmental impact, resource competition, population growth, and food waste, loss, and quality have also been raised and could not be disregarded in addressing the rapidly growing future food demand. To overcome these challenges, SF is widely used and it is a farm management concept to facilitate the sustainable intensification of food production. In SF, the farming process is continuously monitored by using modern IoT and ICT technologies, as well as new paradigms such as the IoNT. The insights derived from the collected data are subsequently used to optimize farm management for sustainable intensification of productivity as well as the food supply chain. Therefore, learning from data plays a crucial role in obtaining timely and useful knowledge that can subsequently be used to increase the success rate of the SF sector. However, several challenges hamper effective data processing, and consequently, the significance of SF data as well as the full potential of the technologies are significantly under-utilized. This PhD research, therefore, aims to explore advanced and effective data analytic methods to increase the potential of making timely and accurate decisions for effective farm management, while overcoming such challenges in the context of smart dairy farming. The thesis has identified five critical challenges which limit effective data analysis and decision-making. Based on these challenges, three research questions (RQ) to address these challenges have been formulated. The RQs mainly aim to simplify large-scale complex data through the development of distributed data mining techniques, and optimizing the energy utilization in data analytics.

Document Organization

This chapter discussed the background and motivation, along with the scope and objectives of this PhD research. In chapter 2, the thesis will present a survey of relevant literature, followed by a summary of the contribution of the research in chapter 3. In chapter 4, conclusions of the present research will be presented, including the future works. Finally, the thesis will present the research outcomes through original research articles in the subsequent Appendices.

Chapter 2

State-of-the-art

In this chapter, we discuss the state-of-the-art solutions that address the research problems considered in the PhD research. Initially, section 2.1 presents an overview of the agri-sector. This is followed by a discussion on the current solutions for data analytics in SF in section 2.2 which includes data collection methods, ML models, and analytical frameworks proposed to overcome limitations and drawbacks. Next, section 2.3 presents the methods for resource management in data analytics. Finally, section 2.4 summarizes data analytics in SF, emphasizing the current challenges that need further attention.

2.1 Overview of the Agri-sector

Agri-sector in the Past

With the industrial revolution in the 19^{th} century, advances such as nitrogen fertilizers and harnessing the energy from fossil fuels and integration of information technology with geospatial tools (e.g., geographical information systems (GIS) and general positioning systems (GPS)) have been introduced into the agri-sector. This has enabled the agri-sector to gain significant improvements in food productivity [2]. However, drawbacks such as high costs and limited data monitoring capabilities have made their deployments unsuitable for large-scale applications in the agri-sector. At the same time, the integration of these new technologies have primarily focused on increasing productivity and paid minimum attention towards mitigating critical issues such as sustainability, diminishing natural resources (e.g., arable land) and uncertainty in climate conditions. These factors cannot be disregarded with the rapid growth of the agri-sector. Ramping up the food production to feed the increasing world population continues to be a critical challenge. The FAO report [18] has emphasized

that major transformations of agriculture systems for food production and supply chain are necessary to overcome these challenges, and to provide secure and healthy food for everyone. Therefore, endeavors for developing sustainable agricultural systems to increase high-quality food products had been a vital concern in the past.

Agri-sector Today

Today, the agri-sector widely uses SF concepts as a promising solution to address the increasing food demand. Under SF, various advanced devices and systems have been proposed into the agri-sector and they contribute to optimizing the farm management process while mitigating the challenges mentioned in section 2.1. Sensors are increasingly being introduced into the modern agri-sector, and almost two-fifth of the farms are now using sensor-based systems (e.g., wireless sensor networks) [19]. The sensors are generally used for monitoring various farm processes in order to create opportunities to optimize farm management practices. Examples of sensors being used include *AfiMilk*, *AgriCam*, and *FullWood* sensors, which are used for detecting mastitis and providing early warnings for producers to take preventive actions. Meanwhile, *SmartBow* and *EmbedVets* sensors are used to monitor cow rumination and mobility behaviors; rumination monitoring enables detecting digestive problems, while tracking mobility patterns provides insight on the animal behavior, which can subsequently be used to detect health issues such as lameness [20]. Various robotic systems are also used in diverse SF applications to replace labor-intensive tasks more effectively and autonomously. For example, the *Lely's Astronaut A5* and *DeLaval's Voluntary Milking* systems do not only cut the labor cost but also allow cows to milk in a comfortable and safe environment. The *LelyGrazeway* is another robotic technology used for cows to guide them to suitable grazing locations [21]. The report in [21] also mentions the use of drones for the agri-sector. For example, *PrecisionHawk* and *Agras MG-1* are used for detecting grass growth and spraying different chemicals such as fertilizers, pesticides, and herbicides; grass growth monitoring and applying nutrients based on their need in order to manage high-quality grass are critical for pasture-based dairy farms as grass is the primary feeding source of cows. Furthermore, more advanced systems formed by integrating sensors and ICT infrastructure are also available today in the agri-sector. For example, *Stellapps* [22] leverages IoT to offer all types of SF services from general herd management to milk evaluation and distribution while *Agriwebb* [23] facilitates farm record-keeping. Also, *KEENAN InTouch* [24] is another integrated system that uses a Cloud-based system to

provide information for proper feed mixing formula to improve efficiencies and minimize cost and waste.

Overall, these advanced devices, systems, and technologies provide a range of opportunities to monitor and track the dynamics in the agri-ecosystem, and to adjust it accordingly to intensify sustainable productivity. The study in [25], for instance, described the general use of these high-tech systems in the SF environments under four main categories which includes monitoring, interpretation, integration, and fully/semi-autonomous decision-making. In [26], an in-depth summary is discussed on the use of robotic milking and automated farming.

The adoption of these high-tech systems is increasing continuously and it is expected that connected devices in the agri-ecosystem will grow to 225 million by 2024 [1]. In many studies such as in [4], the collaborative use of these high-tech systems is highly recommended as it improves the potential of empowering new services. However, achieving such a level of integrated functionality is still at the early stage, and there are several challenges such as lack of infrastructure, functional incompatibility, and limited inter-communicability. As a result, today, these systems operate in isolation and that their full potential is significantly under-utilized in SF practices [27]. Therefore, the agri-sector requires advanced tools and technologies which are capable of performing intelligent analysis while optimally and effectively utilizing the available resources.

Agri-sector Tomorrow

As SF technologies continue to develop, the agri-sector of the future will be equipped with advanced high-tech systems that seamlessly work together as a smart web to enable fully automated SF practices. Such advanced systems will not only enhance monitoring, controlling, and optimizing farming and supply chain practices but will also encourage more advanced and intelligent agricultural systems. This will create opportunities for producers to adjust their farm management practices based on facts rather than gut feelings [3]. Consequently, SF systems will be more proactive, which in turn, enables operating demand-driven farming with improved end-to-end visibility and traceability in order to ensure information such as quality and safety of food in the supply chain. Furthermore, they will open paths for improving customer food awareness and also reduce food scandals such as the distribution of low quality and fake food brands. Therefore, to achieve these levels of advanced and autonomous agricultural systems in the future, we discuss the attempts that have been made to propose novel solutions as well as their gaps in the context of data analysis.

2.2 Data Analytics in Smart Farming

The growing prevalence of high-tech systems in SF facilitates the collection of large data sets that can provide unprecedented decision-making capabilities, and not only for farm management but also for the entire food supply chain. The study in [28], for instance, emphasized that data has become one of the most valuable assets in the agri-sector and analysis of data is necessary to make important decisions. Moreover, among the several application domains that acquire direct benefits to enhance the quality and efficiency of their services through data analytics outlined in [29], the agri-sector is one of the domains that are greatly emphasized. Therefore, the analysis of SF data using advanced AI and ML techniques to extract insights plays a vital role in the SF decision-making process.

2.2.1 Data Generation

As mentioned in the previous section 2.1, numerous systems have been proposed to monitor various aspects of the SF process in different ways. The study in [30] discussed farm monitoring systems under three main methods: satellite-, images-, and location-based monitoring systems. Examples for these technologies are remote sensing-based drought index to monitor the arid and humid areas using multi-sensor satellite data [31], a drone-based image system for crop monitoring [32], and a multi-sensor system based on five sensor modules for collecting crop canopy traits in agri-terroirs [33]. Similarly, the study in [34] described three main data monitoring mechanisms used in SF, which are machine-generated (e.g., sensors, drones), process-meditated (e.g., commercial data such as purchase behavior), and human-sourced (e.g., photos, videos, books). The recent advances in nanotechnology have also made a notable contribution to the development of data monitoring systems, enabling the collection of data that previously could not be sensed such as deep inside the animal body. This has resulted in IoNT devices such as affinity-based nano-sensors that expands data monitoring scales from the macro-scale to nano-scale to collect data at the molecular level. The study in [5] discussed several applications of nano-sensors in SF such as monitoring fertilizers, herbicides, and crop and grass condition.

These monitoring systems collect heterogeneous data from diverse sources and then form different types of datasets. According to [35], they are mainly time-series, streaming, sequential, graph, spatial, and multimedia datasets. In modern-day data science, large datasets are commonly characterized in the Big Data V's, and so far, there are six types; Volume-large in size, Velocity-rapid data generation, Variety-heterogeneity, Veracity-quality, consistency and trustworthiness, Variability-different rates in the data flow, and Value-diversity in meaning

[36]. Different studies have, however, considered different number of V's such as 3Vs, 5Vs and 6Vs to characterize their data. For example, the study in [6] used the first 5V's to explain big data, including the challenges associated with them and possible solutions. Moreover, the study [27] explored the use of these data types particularly along the first 3 V's in the smart dairy farm domain under seven different categories (dairy farm, feed, breed, health, food, retail, consumer), and each category was further sub-divided as animal, farm, and network of farms. This study concluded that time-series type animal-based data has been taken into account at the farm-level in numerous studies followed by multimedia data. Moreover, this study concluded further that data volume has notably been taken into consideration compared to Velocity and Variability.

2.2.2 ML Techniques and Models

Various ML models have been proposed for performing different analytical tasks based on the collected data in order to make decisions. Hidden Markov Models, Regression Models, Support Vector Machines (SVMs) and Artificial Neural Networks (ANN) are a few examples. The study in [37] developed the SVM and ANN models for long-term weather prediction and concluded that the SVM model performs better than the ANN. In another study [38], a Hidden Markov Model was developed for studying animal mobility and used for detecting atypical behavioral dynamics in group-living animals. Similarly, fuzzy-logic based models have been proposed to detect abnormalities (e.g., nutrient deficiencies) in dairy cows based on the variability in milk properties [39]. Greenhouse gas emission is a critical issue in the SF sector and the study in [40] proposed a regression model to predict methane emission. Moreover, multiple linear regression (MLR), adaptive neuro-fuzzy inference system (ANFIS), and NN have also been proposed to estimate the dairy grassland biomass in [41]. The model performance analysis confirmed that the ANFIS model performed better in predicting the biomass compared to the other two models.

The use of Deep Learning (DL) for complex data analytics has recently shown great potential in achieving state-of-the-art performance compared to the ML models described above. The review in [36] provided a comprehensive discussion on DL models and their applications, including the SF-sector. For example, this review reported that a Convolutional Neural Network (CNN) proposed for plant disease recognition can identify thirteen different categories of plant diseases with 96% accuracy. This study discussed another CNN model proposed for obstacle detection in farmlands for automated machine movement. Similarly, a detailed discussion on the application of different DL models for classification have been

proposed for various SF applications in [30]. Examples of DL models used for classification application includes the Long Short-Term Model (LSTM) proposed for crop and plant leaves classification and the Inception-ResNetCNN model for identifying tomatoes. Although DL-based techniques are promising solutions for processing large-scale complex data and achieving high performance, drawbacks such as the need for vast amount of resources and large datasets for training make their deployment unsuitable, particularly in resource-constrained SF systems that utilize WSN. However, the study in [36] highlighted techniques such as network compression that are suitable for resource-limited devices such as IoT sensor devices.

The combined use of ML models and different data mining techniques has been widely considered in many applications to overcome limitations when processing complex data. A number of studies has been conducted to mitigate the challenges raised due to the unprecedented volume of data. Many studies presented evidence that the representative learning methods have widely being used, where they are able to derive a common and meaningful representation for large-scale complex data by using techniques such as dimension reduction, compression and feature selection. For example, the study in [42] explained unsupervised clustering and principal component analysis (PCA)-based dimensionality reduction techniques in combination for reducing the data size to select dairy herds with similar performance characteristics. Similarly, the Kalman filter-based technique was developed to extract only meaningful data in [43]. The aim of developing this method was to overcome communication overhead with data transmission in WSN-based SF applications such as weather prediction and crop disease prediction. The study in [44] proposed a joint method of discrete wavelet transform (WT) coupled with a multivariate regression method to analyze large-scale data effectively and then proved that WT compressed data can achieve similar accuracy as the original data. Meanwhile, the work discussed in [45] used WT and PCA together to extract salient features from large image datasets. Moreover, the combined use of DL models with conventional ML techniques such as PCA and partial least square regression (PLSR) has been considered for large-scale image processing applications and proved that the combined use of ML models enhances learning accuracy, mitigating the limitations when these models are used in isolation [46]. However, there are no evidence that exists to show that the joint model techniques have been used for SF applications.

Furthermore, data fusion methods have also been used widely for integrating data coming from different sources to extract insights effectively. Most specifically in SF, sensors in WSN produce highly redundant and noisy data, and data fusion techniques have been used to reduce the volume of data by removing redundant data and improving the accuracy of

learning outcomes. For example, the study in [47] used data fusion models to aggregate noisy data coming from wireless sensor nodes to explore limitations in sensing coverage. Based on the data analytic outcomes of this study, data fusion contributed to improving the coverage of WSN. Meanwhile, to reduce the volume of data sent to the Cloud, the study in [48] proposed a technique based on data fusion at the data-level and decision-level, which optimizes the energy consumption. However, fusion techniques are not widely used as they are signal-specific and developing different techniques for various signal types is not feasible.

In addition, transfer learning, active learning and kernel-based learning are amongst the numerous learning methods and the study in [6] documented the use of these methods in terms of overcoming the big data issues associated with Big Data V's. The study in [27] also considered the use of different learning methods under four categories as supervised learning, unsupervised learning, semi-supervised classification, and reinforcement learning. Supervised learning techniques have been reportedly used to deal with various data analytical applications, and classification is the most commonly used among the supervised learning method. However, the ML models and methods have mainly been proposed for various single farm usage and are mostly application-specific and limited in scalability and reusability. These drawbacks limit their cooperative applications, while managing their complexity and flexibility. Therefore, necessary modifications are essential to use them cooperatively in large-scale SF applications.

2.2.3 Data analytic platforms

In order to perform data analytics for making timely and accurate decisions, various data analytical platforms have been developed by using the ML models and techniques discussed above. These platforms are based on the Cloud, Fog, and Edge computing paradigms.

The study in [49] proposed a Cloud-based analytical platform for herd management in which data collected from multiple farms is aggregated at a central server to derive insights that can subsequently be used to improve cattle herd management, such as maintaining optimum animal well-being. However, it is essential to gain more in-depth knowledge about the inter-relationships between different parameters to optimize farm management [7]. The study in [50] developed a system based on an ontology model for cattle management and is more suitable compared to the platform proposed in [49]. That is because it considers both the environmental and livestock parameters and the ontology model converts such data into high-level context information that can be used to understand significant relationships between the parameters used for decision-making.

The Fog and Edge computing paradigms operates in conjunction with the Cloud and processes data that are in close proximity to the sources to perform real-time decision-making. This form of data analytics minimizes the transmission of data to the Cloud and lowers the energy depletion of the sensors. The study in [10] proposed a Fog computing-based IoT system (*AgriFog*) for SF and concluded that it increases the decision-making efficiency with reduced latency. Since most farmers cannot apply the analytical outcomes produced by these advanced systems due to lack of understandability and interpretability, authors in [51] developed a more user-centric analytical framework based on the Fog and Edge computing paradigms by considering the requirements of the agri-producers. The study in [52] proposed a novel analytical approach called *Future IoT controller* in which Edge computing is used with a Bayesian network in order to generate useful intelligence from data. Although the use of DL in Fog and Edge computing-based applications is challenging due to high resource consumption, authors in [53] proposed a distributed deep NN (DDNN) framework for training a DL model over Fog, Edge, and Cloud devices in a distributed manner. The DDNN framework can perform learning more effectively, and this is because a set of layers of the DL model are trained over distributed Fog or Edge devices, while the remaining layers are trained in the Cloud. Thus, this framework reduces the amount of data transferred to the Cloud, which in turn, increases the efficiency of the data analytical process.

Moreover, several studies report that WSN-based data analytical platforms integrated with Cloud, Fog and Edge computing paradigms have been proposed for various SF applications. For example, the study in [54] proposed a framework that integrates specialized electronic sensors for autonomous collection of on-site soil and climate data (e.g., humidity, soil temperature, and rain gauge) in a distributed manner. The system then processes the data to provide real-time insights to validate various biological and ecological models. Similarly, authors in [55] also proposed a WSN-based decision support system named *iFarm* to provide decisions, aiming to enhance crop productivity. *ViSeed* is another ML and visualization framework that was developed for predicting crop yield and the weather [56]. Also, a multi-sensor node based intelligent decision support systems for irrigation management is presented in [57]. The ML algorithms developed in these decision support systems use the real-time data collected from sensors to control the delivery of water, aiming to reduce wastage of freshwater in farming activities.

A comprehensive discussion on the distributed analytical systems given in the study [58] outlined that the Fog and Edge computing paradigms based analytical platforms and their benefit for distributed and real-time data analytics. However, the need for fully decentralized analytical platforms has been raised due to concerns from data sharing and the

lack of guaranteed transparency and validity of learning outcomes. Alternatively, this study discussed further on the use of FL and BC methods that can contribute to overcoming such drawbacks while ensuring the decentralized data ownership. For instance, the study in [59] proposed a FL-based mechanism to predict hospitalizations due to heart disease by securely incorporating multiple health records sources. A SVM classifier-based approach was used to build the prediction model and the study concluded that the proposed approach could achieve similar accuracy as standard CML approaches. This study highlighted further that effective communication of model updates between clients and the coordination unit play key role in FL-based systems for improving the efficiency of decision-making. A NN model-based FL system discussed in [60] proposed two models for improving the communication of model updates and they are structured model and sketched model. While the structured updating method considered updating from a low-dimensional data space (e.g., PCA-based learning), the sketched updating method was based on compressing model updates before sending it to the coordination unit where the final model refinement is carried out. However, the study in [61] highlighted that the application of FL-based systems is feasible only when there is a trustable coordination unit, including some other drawbacks such as security risks, P2P interaction and functional failure of the coordination node that limits the use of FL-based systems. An alternative solution is the incorporation of Block Chain (BC) to overcome these limitations.

Several application domains have taken into account the integration of Cloud, Fog, and Edge computing-based analytical platforms with BC technology. This is not only because of its capability in overcoming issues in FL-based systems and improving learning accuracy but also its flexibility in ensuring several features such as transparency, validity and traceability, which are essential in employing collaborative decision-making. For example, to improve the accuracy of medical diagnosis and treatment efficiency, a BC-powered parallel healthcare system was proposed in [62]. This system has the potential to securely incorporate information coming from various healthcare communities to enhance treatment efficiency. The main advantage of this parallel healthcare system compared to the FL-based healthcare system proposed in [59] is that single point functional failure is completely controlled. The studies in [63, 64] also proposed BC-enabled systems for the transportation and energy markets in order to make their services more effective and smart. More applications of BC-enabled decision-making systems can be found in [65].

Considering the attempts that have been made to develop BC-enabled systems in the SF domain, the study in [66] proposed a BC and IoT integrated system to make agri-products traceable. However, this study did not provide any experimental evidence to validate their

system. Moreover, Nori [67] and Regen [68] are consortium BC-based (collection of BC networks) platforms mainly proposed for improving the sustainability of the agri-sector. While Nori was developed for reversing climate change through reducing CO_2 emission, the Regen network enables monitoring ecological degradation and climate change. The study in [69] also discussed the use of BC technology in the agri-sector terming it as *E-agriculture*, where discussions revolved around opportunities, benefits, and challenges of using BC in agriculture. Moreover, the *AgriDigital* is an expertise in the use of BC in the agri-sector and emphasizes the main aim of using BC to improve traceability in agri-food supply chains [70]. However, compared to the attention that has been given for applying BC in other sectors, it is notably less in the SF sector.

There are several challenges such as security threats and resource scarcity that still requires further investigations in integrating BC into existing systems. The studies in [71, 72] discussed these challenges in detail. However, some studies have warned that such BC-integrated systems could bring negative effects. For instance, the study in [69] warned that this integration can also lead to unnecessary computational overhead and may not bring any tangible benefits. Therefore, the studies in [65, 69] recommended that conducting an initial case study to make sure that integration with BC is necessary by proposing a checklist to conduct such a feasibility study.

2.3 Resource Management for low-powered devices

Resource scarcity is a critical challenge for distributed data analytical systems, particularly in resource-limited IoT infrastructures. To optimize the resource utilization in distributed computing platforms, various in-network computing techniques (e.g., cooperative computing, parallel computing) and methods of harnessing energy from natural resources (e.g., solar, wind) have been proposed.

The study in [73] explored the performance limitations in dynamic resource allocation to achieve the maximum query rate with distributed data processing. The performance limits taken into account were primarily concerned with data volume and velocity under the resource-constrained distributed processing environments. In order to overcome these limitations, an algorithm was proposed that can dynamically allocate computational and network bandwidth resources. Meanwhile, The work in [74] presented an approach to optimize total energy consumption cost in mobile WSN. This approach enables optimal partitioning, offloading and execution of tasks cooperatively with peer nodes. In addition, a cooperative computing node selection mechanism was also presented with a particular

emphasis on the fair selection of nodes to avoid each node's energy being over-used. IoT-based systems incorporating wireless sensors have not only been proposed for collecting data but also performing distributed computations to improve the overall efficiency of their services [75]. The approach proposed in [75] discussed computing in a two-node network, block computation over multiple nodes, distributed computation with noise, along with randomized algorithms for distributed computing.

Data redundancy and replication can also increase the amount of transmitted data over distributed networks, which will demand high bandwidths. To overcome this issue, offloading data as segments to neighboring devices was taken into account in [76], aiming to improve the probability of successful data delivery. A probabilistic framework-based heuristic algorithm was proposed to estimate the probability of data delivery through an opportunistic path by considering the parameters data size and contact duration. The experimental outcomes proved that cooperative offloading could improve the probability of data delivery. This study also proposed a distributed algorithm that aimed to cooperatively offload data in a distributive manner. Meanwhile, the DDNN framework [53] mentioned earlier trains a DL model as segments over a distributed Fog computing network and proved that it could achieve 20 times less communication cost compared the standard Cloud-based CML approach. This study emphasized further that the involvement of remote Cloud in such distributed analytical processes raises data privacy and ownership issues and limits cooperative computing. As an alternative to these issues, the study [77], for instance, proposed a design of a cooperative computing system that includes only mobile devices and there is no involvement of a central Cloud. This system was termed as *Serendipity* in which computation tasks are sub-divided and shared between mobile devices in order to minimize the task completion time.

Although the study in [75] emphasized that WSNs are widely proposed for both data monitoring as well as performing application specific distributed computing, end users (e.g., sensors) mostly suffer from lack of energy since most of the sensors are battery-powered. As a solution, harvesting energy from natural resources such as wind and solar power has been widely proposed [78]. However, the uncertainty in natural energy sources limits the deployment of such energy harvesting devices. To overcome this uncertainty, the study in [79] proposed a hybrid framework that combines solar power and wireless charging. A distributed algorithm was proposed for optimally deploying solar-powered cluster heads and exploring energy balancing in the absence of solar energy, while a polynomial-time scheduling algorithm combined the solar power and wireless charging methods for mobile data gathering. The study further claimed that the hybrid framework could reduce battery depletion by 20%, including the 25% of system cost compared to the previously proposed

wireless-powered systems. Another study in [80] compared the performance of the energy harvesting relays against the conventional cooperative relaying in wireless communication, and found that this architecture has better energy efficiency compared to the conventional relaying network and could be a promising solution for energy-constrained wireless networks. A study similar to [80] presented in [81] also discussed the resource management issues in decode and forward relay based cooperative IoT networks and proposed a mathematical model that aims to maximize the data communication rates.

2.4 Summary: Challenges and limitations

This chapter primarily considers the research contributions towards the development of ICT based agri-sector in terms of data collecting, processing, and analyzing, with an emphasis on their roles in smart dairy farming. With the advances in IoT and ICT technologies along with data science, more intelligent and effective ML models and data analytical systems have been introduced into the agri-sector. They have made a immense contribution to the sustainable intensification of farm productivity to meet the increasing food demand.

However, many studies claim that there remains critical challenges that need further attention. In particular, the unprecedented volume, high complexity and dynamic nature of data challenge the processing capabilities of existing ML models and data analytical systems (**C1** and **C2**). Also, these models and systems are often poorly coupled and reused due to several reasons such as lack of interoperability, scalability, and accuracy [82] (**C3**). Similarly, the study in [83] also highlighted that data ownership and privacy, data quality, intelligent processing and analysis, sustainable integration of data, business models and openness of analytic platforms are the most critical issues associated with the development of harmonized analytical frameworks in the agri-sector (**C3** and **C4**). The study in [84] argued that sometimes resource scarcity for development, evaluating, and applying agri-models combined with lack of knowledge on user-centric models creates greater challenges than these critical issues (**C5**). All in all, the full potential of the data as well as the modern SF technologies is significantly underutilized. Therefore, the SF sector urgently requires novel ML models and analytical frameworks for processing data effectively.

The study in [27] made an overall evaluation considering nearly 1500 different studies conducted for analyzing big data over the period 1994-2017 for the smart dairy farming. This evaluation concluded that the majority were related to mitigating issues related to the volume of data, followed by the variety and then the velocity, while the attention on other V's was much less (**C1**). This study also concluded that the full potential of big data collection in

the dairy farming sector is not fully utilized. Meanwhile, the study in [7] emphasized that most state-of-the-art models and systems available today in the dairy farm domain have been used at the individual farm-level. Hence, more efforts are essential to use them cooperatively by connecting several farms. As a most viable solution, the work in [6] suggested that ML models and learning platforms have to be re-shaped based on the application requirements (**C2** and **C3**). Moreover, authors in [85] emphasized that more integrated data analytical platforms that can produce more intelligent decisions is still very ambitious (**C4**). The study in [4] also outlined that the services of the decision support systems available in the dairy domain can be enhanced by collectively using them in the farm management process.

Chapter 3

Research Summary

This chapter describes the solutions that have been proposed to address the research RQs, overcoming the challenges discussed in chapter 1. First, section 3.1 presents the solutions that have been proposed to answer the research RQs. Second, the tools and experiments used for validating these solutions are presented in section 3.2, followed by the contribution of the present study to the data analytics in the SF sector in section 3.3. Finally, section 3.4 presents the answers for the research questions.

3.1 Research Tasks

In this section, this PhD research addresses the three research question sequentially under five tasks as illustrated in Figure 3.1, and each task is based on findings from the previous tasks.

RQ1 - Task 1: Compressed Learning (CL)

This task explores the mechanism of data compression and how it can contribute to simplifying complex data for effective learning. First, this task examines the drawbacks of existing data compression techniques in addressing the challenges induced by voluminous data and then propose our solution. In fact, the existing compression techniques such as *Lempel-Ziv-Welch (LZW)* [86] are able to perform zero data loss compression and can optimize storage requirements. However, their main drawback is that the decompression is necessary for applying data analytics on data and consequently, it restores the original data complexity during the analysis phase. The reason is because the compression is performed without minimizing the complexity of the data by removing insignificant and redundant data. As a

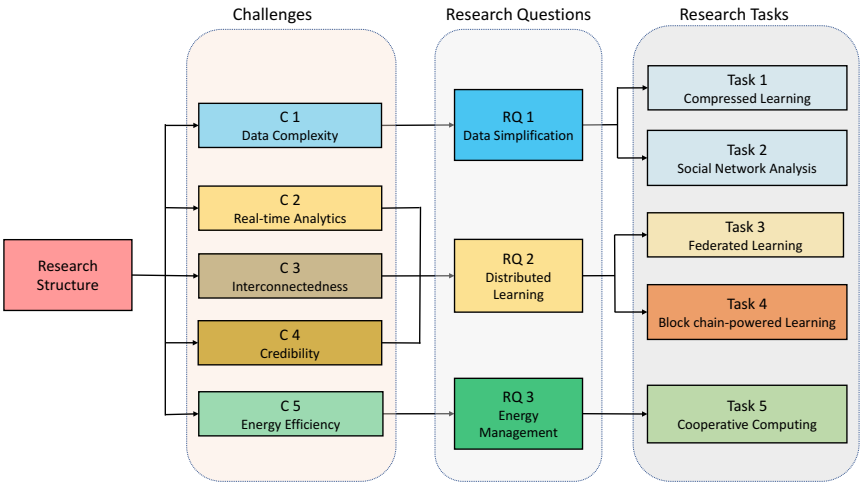


Fig. 3.1 An overview of the research plan and the mappings between the challenges, research questions and research tasks.

result, this makes their deployment unsuitable, particularly in data-intensive, time-sensitive and resource-constrained big data analytic applications. To overcome these drawbacks, this task proposes a solution named **Compressed Learning (CL)** (Appendix A and B).

CL is a concept whereby learning is conducted in the compressed data domain of the original data, while preserving the original accuracy. The compressed data can also be obtained back by using recovery algorithms [87]. There, only the learning outcomes such as empirical models and their performance coefficients, along with the recovery algorithm, are sufficient for the processing engine analytics to make decisions. The CL approach encourages compressing data in proximity to the source using the domain-specific knowledge and then performing learning on the compressed data. The algorithm also derives optimal compression parameters and stores them with the recovery algorithms if decompression becomes necessary for further analysis in the future. Therefore, CL enables data analytics with small datasets as well as reducing data processing, communication and storage overheads in large datasets. Furthermore, CL prevents restoring the original complexity of the data during the learning process, thereby contributing to significant improvements in both computational and statistical efficiency.

The use of CL for SF is demonstrated by analyzing Mid-Infrared Spectroscopy (MIRS) of milk for assessing milk quality. Analysis of the quality of milk is crucial as the milk is the main product of the dairy farming industry and conveys valuable information such as animal nutrient deficiency and health issues which are helpful for efficient dairy farm management. Although the transfer of MIRS datasets from farms to a central Cloud leads to collection of precise information, it is not always feasible due to reasons such as the intermittent

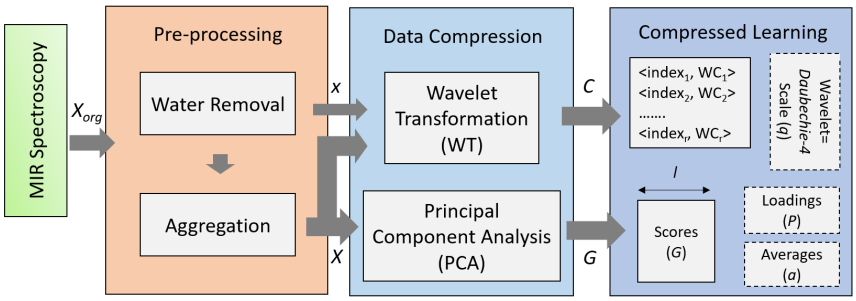
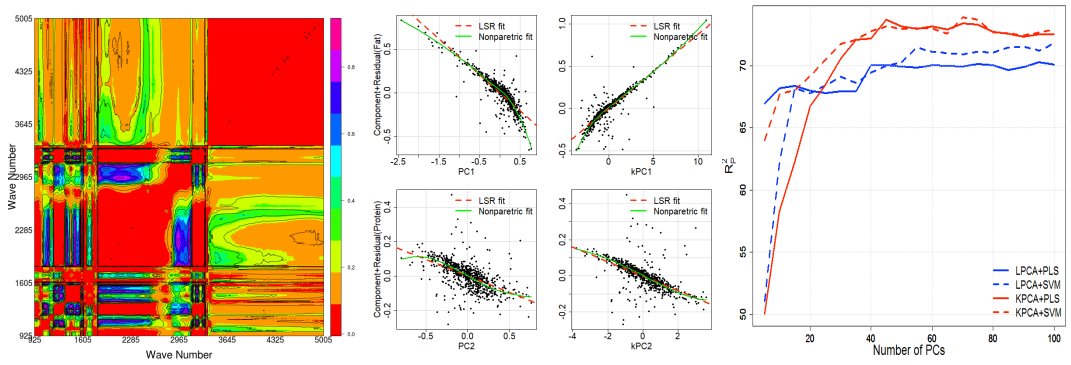


Fig. 3.2 Compressed Learning pipeline (Appendix A, Fig. 2 in page 6)

broadband connectivity in rural farms. In this context, CL can be leveraged under the Fog or Edge computing framework, thereby minimizing the resources required for communication, computing and storage while improving application latency. Therefore, this task explains the process of deriving near-lossless compression parameters to transform MIRS data without impacting the prediction accuracy for a selection of Milk Quality Traits (MQT). They are Lactose, Fat, Protein, and Urea and are used as the main MQTs to decide the quality of milk based on their composition. Moreover, this task emphasizes the significance of using domain-specific knowledge in the CL process.

The CL pipeline is illustrated in Figure 3.2. Initially, a comprehensive pre-analysis is conducted to understand the original characteristics of the dataset and then selected compression and learning techniques accordingly. This helps in not only preserving the original features of the dataset but also avoids the use of computationally heavy ML techniques. Based on this, unnecessary data in the MIRS dataset is identified and then removed. In this PhD research, we focus on the data corresponding to water in milk. To understand different characteristics of the dataset, this study explores the various relationships between the feature variables in the MIRS dataset as well as the feature variables and response variables. The relationship is multi-collinearity and includes both linear and non-linear correlations (see Figure 3.3). Based on these characteristics, Principal Component Analysis (PCA) and Wavelet Transforms (WT) are used as the linear compression techniques, while the Kernel PCA (KPCA) is used as the non-linear compression. This combination enables compression techniques to integrate data analytics on the compressed data. In the next stage, two learning techniques which are the Partial Least Squares Regression (PLSR) and Least Square Support Vector Machine (LSSVM) are used for linear and non-linear learning.

The MIRS data analysis confirms that MIRS data can be pre-processed and compressed effectively near the data source without impacting on the prediction accuracy of most measured milk quality traits. The pre-analysis reveals that the removal of redundant and unnecessary



(a) (b) (c)

Fig. 3.3 Linear and Non-linear relationships in MIRS data and CL performance; (a) difference between Linear and Non-linear correlation between feature variables in MIRS dataset, (b) non-linearity between Fat (first row) and Protein (second row) milk quality parameters with MIRS dataset, (c) linear and non-linear performances for Protein (Appendix B, (a)-Fig. 3 in page 5, (b)-Fig. 6 in page 7 and (c)- Fig. 7 in page 7).

data contributes to achieving a 20% compression rate. While the linear compression by PCA and WT achieve 92% and 91% compression rates, the non-linear compression from KPCA attains 89% compression rate. At these compression levels, PLSR and LSSVM can achieve similar accuracy as the original data domain. This justifies how the original learning accuracy can be achieved with significantly less feature variables using CL. Therefore, transferring only the compression parameters corresponding to these compression levels along with recovery algorithms to a central server is sufficient for further analysis or long-term storage of data. In addition, the comparison of CL performances with advanced Deep Learning techniques such as ResNet, LeNet-5 and Vgg-19 reveals that the CL performances are comparable for certain MQTs, but it is typically less compared to other techniques. However, the high resource consumption of these Deep Learning models makes their deployments unsuitable for resource-constrained environments like SF. Therefore, in this context, CL is a good alternative and can effectively be used for performing real-time data analytics in SF.

RQ1 - Task 2: Deriving Novel Metrics

Recent developments in advanced data monitoring technologies using WSN, facilitate collection of vast amount of data and requires novel techniques to transform data into useful metrics. This task considers deriving useful and informative metrics from sensor data and use them for decision-making. As a use case, the social behavioral dynamics of dairy cows are explored based on their mobility data. In general, cows are gregarious animals and have

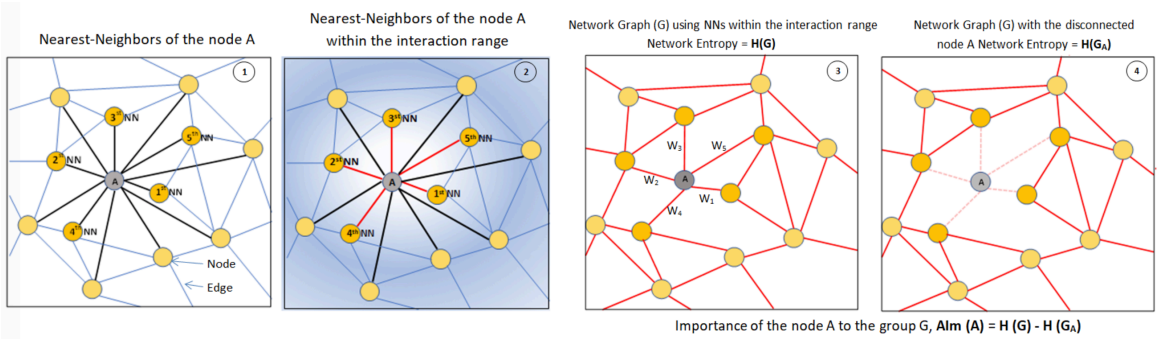


Fig. 3.4 The steps for deriving *AIm* (in this example we focus on node A); (1) compute the Nearest-Neighbor (NN) of node A, (2) select the NNs in the interaction region, (3) compute the weighted centrality of A and then the network entropy based on all node centralities, (4) *AIm* of the node A is the difference between the network entropies computed with and without A in the network (Appendix C, Fig. 1 in page 4 and Fig. 2 in page 5).

complex social interactions that convey valuable information such as animal well-being, which can be capitalized for optimizing farm management. However, evidence that supports the significance of their social relationships is considerably under-utilized due to the lack of metrics that can effectively characterize underlying dynamics in the data [88].

Social Network Analysis (SNA) is a commonly used technique to explore behavioral dynamics within a social group [89]. An open question in SNA, is whether it is possible to characterize an individual node level behaviors by using the network level group information. The reason for extracting this relationship is because it is difficult to keep track of each individual or small sub-groups as the group size increases. Therefore, identifying individuals that deviate from the normal group behavior is an effective method for decision-making and this can be used for generating early alerts for the prevalence of diseases. However, the existing SNA metrics are capable of only using intermediate-level information (i.e., up to the 2^{nd} nearest-neighbor interactions of a node [90]), and there are no measures which are capable of incorporating group-level social interactivity information for deriving node-level social dynamics. Therefore, in this task the PhD research proposes the **Animal Importance (*AIm*)** metric to compute the importance of a node to a group by incorporating node- and network-level SNA measures (Appendix C).

A SNA-based graph-theoretical approach is used to derive *AIm*, including two supportive metrics, which are the *social interaction range* (k) and the *nearest-neighbor matrix*. These two metrics are used to translate social interactions into a network graph. A network graph illustrates the structural connectivity of a social group and can integrate various information that exhibits heterogeneous social interactions. Therefore, including only the most relevant interactions is crucial for simplifying the network complexity and also to improve the network

graph representation without missing important information. The metric k stands for the range that a node can make significant interactions and is derived by using a topological distance-based method instead of using the traditional metric distance. The topological distance-based method uses the anisotropy value that represents the effect of interaction among animals and quantifies to what extent the spatial variation of nearest-neighbor around a reference animal is anisotropic [91]. This is subsequently used to determine k, regardless of the distance between the animals. The nearest-neighbor matrix enables identifying preferential members (i.e., social affiliation) of every node by computing the frequency of each node being in every other node's interaction range. Therefore, these two metrics can help to select the most significant interactions of each member in a social group, and thereby, enhancing the quality of the structural information acquired for characterizing different social behaviors.

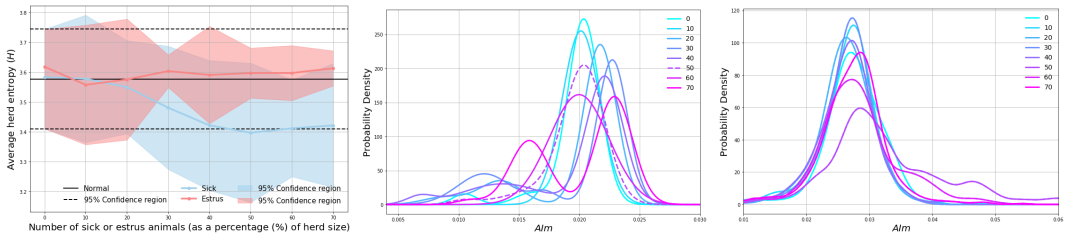
The *AIm* metric is then derived by combining the node- and the network-level SNA measures, node centrality and network entropy. The node centrality quantifies to what extent a node is surrounded by other nodes, while the network entropy depicts the amount of information encoded within a network and is used to compute the structural connectivity at the group level. The network entropy of a group G ($H(G)$) is computed using the node centrality measure, and is represented as follows:

$$H(G) = \log\left[\sum^{N} C_w(A)\right] - \sum^{N} \frac{C_w(A)}{\sum^{N} C_w(A)} \log\left[C_w(A)\right],$$

where $C_w(A) = \sum_{i=1}^{n} w_i$ is the weighted centrality of a node A, w is the interaction strength computed as the reciprocal of the distance between two nodes, and N and n are the group size and number of direct interactions of the node A. Based on this, the influence of the node A on changing the network entropy is considered as its importance to the network ($AIm(A)$) and computed as the change in group entropy in response to the disconnection from all interactions of the node A within the network.

$$AIm(A) = H(G) - H(G_A),$$

where $H(G_A)$ is the network entropy without the node A. A graphical overview of the four main steps of the *AIm* derivation process is illustrated in Figure 3.4. Since social behaviors are highly dynamic due to the time-variant nature of the network topology, we explore the variability in *AIm* at the node-level as well as the network-level by computing the probability distribution function (PDF) of each node as well as joint PDF of *AIm*. The Gaussian Mixture Model (GMM)[92] is used for deriving the joint PDF.



(a) (b) (c)

Fig. 3.5 The application of SNA for determining the number of sick and estrus animals as a percentage of herd size, (a) variability in group entropy, (b) joint PDF of Animal Importance (AIm) for sick animals, and (c) joint PDF of Animal Importance(AIm) for estrus animals (Appendix C, Fig. 11 in page 13).

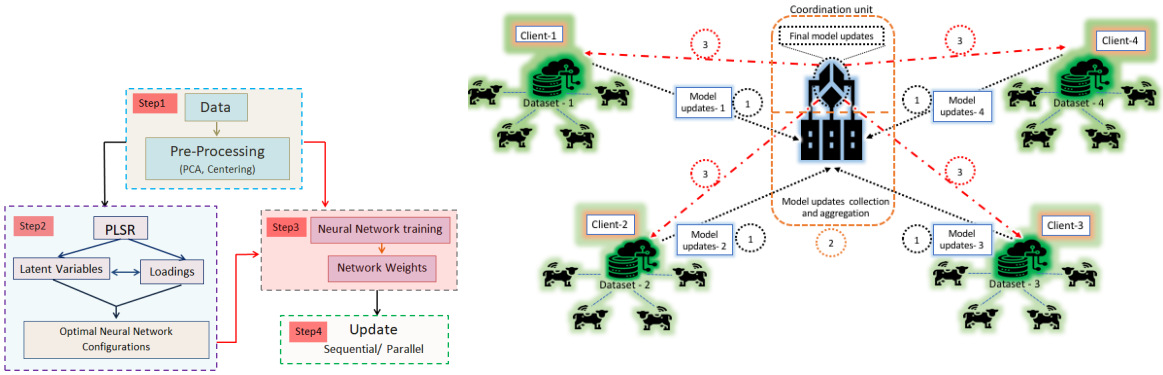
This study evaluates the performance of *AIm* and compared it with already existing SNA measures, which are Weighted Degree centrality and Laplacian centrality [90]. These two measures use local and intermediate level connectivity information for computing cohesion of nodes in a social group. In terms of variability of cohesion over a period of time, *AIm* demonstrates more stability compared to the two existing measures. In order to explore the usability of *AIm* in decision-making, this task demonstrates how *AIm* can be used to detect the sick and estrus cows in a dairy herd based on the evolution of the PDF of *AIm* of each cow in a herd (the experiment is discussed in the next section). For normal cows, the PDFs of *AIm* distribute around a similar mean, but with different variances. Meanwhile, the PDFs of *AIm* of sick cows shift towards the left of the mean of the normal cows PDFs and their means are also not comparable. On the other hand, the PDFs of *AIm* of estrus cows shift to the right of the mean of normal cows, but their means do not vary as much as sick cows. At the group level, we consider the evolution of the joint PDF of *AIm* of all cows and the herd entropy. As shown in Figure 3.5, the joint PDF of the herd with the increasing number of sick cows (as a percentage of herd size) exhibits a bimodal distribution with a left tailored shape, where each line corresponds to the percentage of the number of sick animals in the herd. The joint PDF of the herd with increasing cows in estrus is opposite to that of the sick cows. Furthermore, in comparing the variability in the entropy of normal herd with the increasing number of sick cows in a herd, the group entropy decreases with larger variances, whereas only the variance of the entropy decreases with increasing estrus cows. Therefore, this task suggests that the metrics k, A, and *AIm* can effectively convert large-scale complex social behavioral data into valuable insights which can subsequently be used for to generate early alerts of atypical social behaviors of farm animals which are useful for effective farm management.

RQ2 - Task 3: Federated Learning Based Distributed Data Analytics

To mitigate the limitations associated with data analytics mentioned in C2 and C3, this task first proposes a ML model considering the drawbacks of current models and introduce a framework for training the proposed model by securely incorporating distributed data sources (Appendix D).

As it was mentioned in section 1.2.1, higher dimensionality, multicollinearity, and non-linearity are amongst the most common characteristics that limit the use of simple ML techniques for analyzing large-scale complex datasets. For instance, the use of Least Square Regression (LSQR) models fails due to multicollinearity for specific data types and this is because the algorithm assumes independence between the feature variables. The PLSR method is a promising alternative to manage multicollinearity in the data but can only capture linear relationships. Neural Network (NN) has the potential of capturing complex relationships that is not possible in LSQR and PLSR models. However, the use of NN in resource-constrained applications is challenging since selecting optimal NN configurations is time and resource-consuming task. Therefore, to overcome these limitations, the present study proposes a joint ML model by combining the PLSR method with NN (we refer to this joint model as NNPLS). The basis for deriving this model is because PLSR model is considered as a single hidden layer feed-forward NN in which the number of PLSR model parameters (i.e., latent variables) is equal to the number of hidden layer neurons. The output layer contains a single neuron, while the number of neurons in the input layer is equal to the number of feature variables feeding into the NN. Therefore, in the NNPLS model, the best NN settings for the optimum number of hidden layer nodes and initial weights are derived through the PLSR method, and the NN is used to train and obtain the optimal NN weights. This relationship is illustrated in Figure 3.6a (more details are provided below).

In the ML frameworks currently in use to perform model training and learning, sharing data with a third-party service such as remote Cloud is a necessary requirement due to number of reasons such as limited resources and deployed on-board intelligence on Edge devices in DML systems. However, data owners are reluctant to do so due to several reasons such as high communication costs and data privacy and ownership issues. The SF producers will usually hesitate to share their data with third-parties without knowing the value that they get in return. This is because large-scale companies (e.g., device manufacturers), which provide services for analyzing data, get the majority of benefits from data compared to the data owners. To overcome such challenges, this PhD research uses the Federated Learning (FL) framework proposed by Google [93] to train our model. FL is a distributed learning approach used to train a common ML model across geographically distributed data sources (clients)



(a) (b)

Fig. 3.6 The FL-NNPLS framework and its application in SF; (a) Block diagram of the NNPLS model derivation and training process, (b) A FL architecture that is applied to smart dairy farming. The steps for the learning process are as follows: (1) Every farm client service collects data (e.g., MIRS of milk samples) and trains a ML model and sends the updated model to the central service unit, (2) The central service unit refines the final model by aggregating the model updates from client services, (3) Client services download the final model updates and then update their local ML models to perform inferences (Appendix D, (a)-Fig. 1 in page 2 and (b)-Fig. 3 in page 6).

without moving data from the source. A coordinating, or central service unit, aggregates the model updates collected from the clients and then computes the final updated model which is downloaded by each client for decision-making (see Figure 3.6b). This procedure is continues based on the availability of data from the clients. This allows the ML model to be trained in real-time and at the same time all participants can access the updated model after every training cycle for decision-making. Moreover, this reduces data communication burden as there is no requirement for clients to be connected to the coordination unit at all times. At the same time, data privacy, security and ownership issues can be mitigated effectively since data does not move away from the sources, while only the model updates are transferred over the communication network.

For a given dataset, the main steps of the NNPLS model derivation and its FL-based training processes is illustrated in Figure 3.6, and steps are listed as follows:

1 **Data pre-processing:** Conduct pre-processing on data by using the CL algorithm described in Task 1 and create an optimally compressed dataset.

2 **NN settings:** Fit a PLSR model on the compressed data and then derive the optimal number of latent variables through cross-validation. The optimal number of hidden layer neurons is equal to the number of latent variables in the PLSR model. Based on

this, the optimal initial weights of the input and hidden layers are the loading matrices of the feature and response variables (i.e., X, Y) corresponding to the optimal number of latent variables in the PLSR model.

3 **NNPLS training:** Start training the NNPLS model with the initial NN weights derived from step 2 and compute the optimal NN weights (i.e., NNPLS model parameters). This is the local model updates.

4 **NNPLS updating:** Final updated global model is computed by taking the arithmetic mean of the local model updates. We perform parallel and sequential FL updating procedures following the steps given under Figure 3.6b to compute the final model.

- **Parallel:** All clients compute their local model updates in parallel, and then the coordination unit aggregates them to derive the global model.
- **Sequential:** All clients contribute to computing the global model sequentially. This process will have one client send its local updates to the coordination unit, while another client that is ready to perform model training, will downloads the global model to trains the local model.

There are advantages as well as disadvantages of using these two approaches. The effectiveness of the parallel FL depends on the slowest client as all clients have to finish their local model updating to derive the final global model. The possibility of the global model being bias to the clients which have higher data generation frequency is higher in sequential FL. However, in this PhD research, we assume that all clients have similar data processing power and contribute equally, but in reality their selection may depend on the application requirements.

The performance of the NNPLS model based FL framework (**FL-NNPLS**) is demonstrated for the milk quality prediction using the MIRS data. The analysis takes into account how well the proposed model can predict the milk quality (i.e., the composition of selected MQTs Fat, Protein, and Lactose). Under the CML approach, the MIRS data analysis confirms that the replacement of the LSQR model by the PLSR model and then the NNPLS model contributes to improving the predictive performances of MQTs, and the NNPLS model represents comparable performances to a state-of-the-art Deep Learning model. Under both the sequential and parallel FL methods, higher predictive accuracy can also be achieved by increasing the number of federation steps (see Figure 3.7). Our analysis confirms further that FL-based approaches can achieve comparable performance to the CML approach in

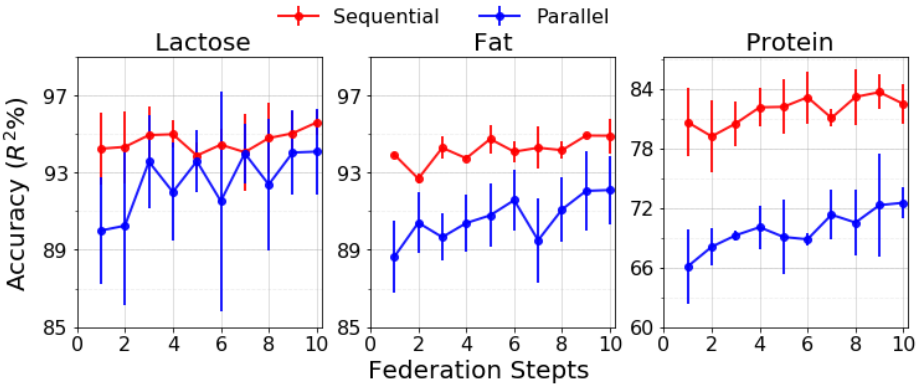


Fig. 3.7 NNPLS model-based FL performance for different milk quality parameters under the sequential and parallel updating approaches (Appendix D, Fig. 8 in page 9).

few federation steps (R^2 values of the three lactose, fat and protein MQTs from the CML approach are 93.39%, 91.12% and 83.09%, respectively).

Additionally, the MIRS data analysis also proves that predictive performances can be improved further by increasing the number of clients who contribute to training the model. However, increasing the clients can lead to a serious issue known as data imbalance. This will happen when there are significantly differing sample frequencies amongst the clients and the learning outcomes tend to be biased towards those who have higher sample frequencies. This in turn will ignore the significance of information coming from clients who produce small samples. The existing solutions primarily suggest that re-sampling (over-, under-, and hybrid-sampling) is a promising way to overcome this issue [94]. However, an additional issue arises due to data imbalance associated with the FL-NNPLS framework. The number of LVs required for the optimal PLSR model varies with the sample size, and as a result, the local NNPLS model configurations can differ among the clients. Therefore, the derivation of the global model by taking the arithmetic mean of local models becomes problematic. To overcome the data imbalance issue, this PhD research proposes a re-sampling and zero-padding based joint approach. While re-sampling equalizes the different sample sizes, zero-padding is used to manage the local model aggregation issue. The joint approach is applied to manage the data imbalance issue and then asses the FL predictive accuracy.

RQ2 - Task 4: Blockchain-based Fully Distributed Data Analytic Framework

Although the FL-based data analytic system has good potential in performing privacy-preserved DML, there are some circumstances that FL may not be the best option, and they are listed as follows:

- **Direct interactivity:** Since data sources (i.e., clients) are identified, authenticated, connected and then communicated through the coordination device, the FL-NNPLS system does not support end-to-end communication. However, end-to-end communication is the key to automating distributed data analytics for producing timely and accurate insights.
- **Single point failure:** Functional failure of the coordination unit can cause the functionality of the entire FL system to collapse.
- **FL Transparency and trustability:** As there is no mechanism to check the credibility of FL clients, their misbehavior can impact the validity of the learning outcomes. For example, the coordination unit could be biased to certain clients and can also ignore or modify data coming from other clients. Similarly, certain clients may pretend that they have sufficient or valid data to contribute to the model updating process and can share invalid information. Therefore, transparency and credibility of the FL process can potentially be challenges that needs to be addressed.
- **Data privacy:** There is still a risk of leaking sensitive data due to the possibility of extracting specific data during the model updates [61].

Based on these challenges, this PhD research considers incorporating Block Chain (BC) technology to develop a fully distributed analytical system. BC is a distributed ledger technology that maintains a shared ledger across distributed data sources or clients. It also allows peer-to-peer(P2P) interaction without involving any central entity to control the system. The ledger consists of a chain of blocks (see Figure 3.8) that store data and are linked in chronological order by using cryptographic keys. These features ensure the data stored in the ledger is secure, transparent and traceable along with no risk of single-point failure. Thus, the integration of distributed analytical systems with the BC technology can contribute towards fully distributed decision-making systems with improved data privacy, security, auditability and transparency. Moreover, advances in nanotechnology can provide further opportunities to improve reliability as well as facilitate new applications. That is because the

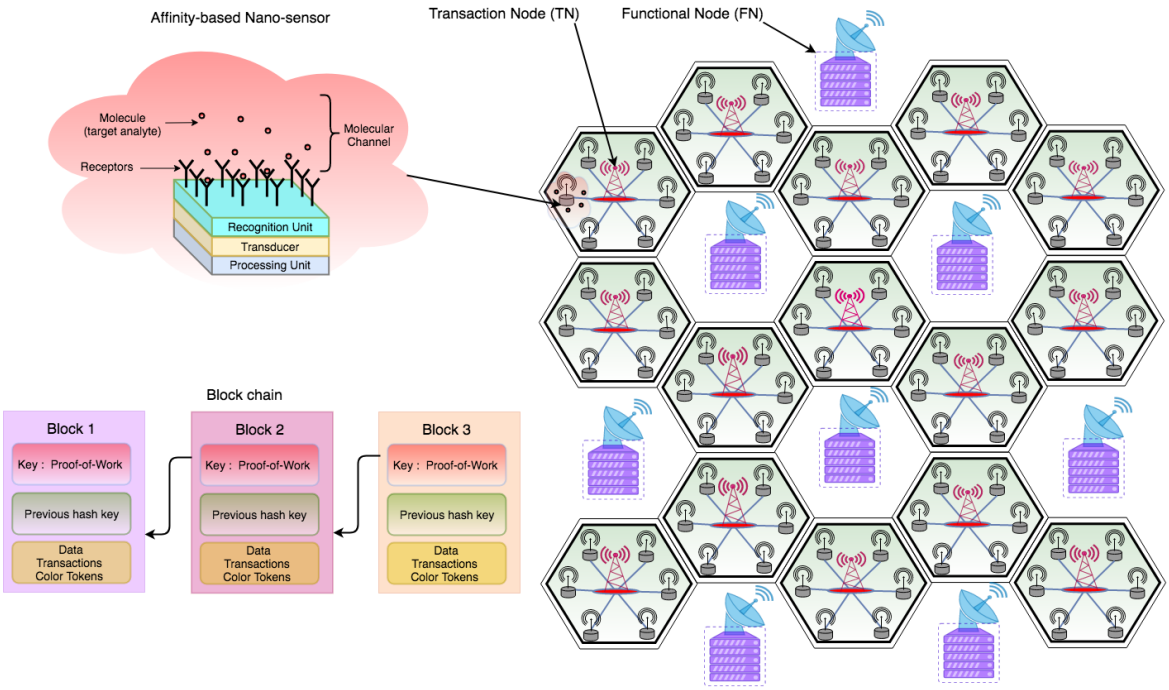


Fig. 3.8 The BC-powered Internet of Nano-Things (BC-IoNT) architecture for detecting level of chemicals in farmlands (Appendix E, Fig. 1 in page 5).

miniature devices that are constructed from nano-materials will be able to sense molecules at fine granular scale, which in turn provides a new spectrum of data that can be analyzed. However, the integration of such devices with the BC-powered decision-making systems in any application domain has not yet been investigated. Therefore, this PhD research proposes a BC system integrated with IoNT (**BC-IoNT**) to employ fully distributed decision-making in the context of farm management (Appendix E).

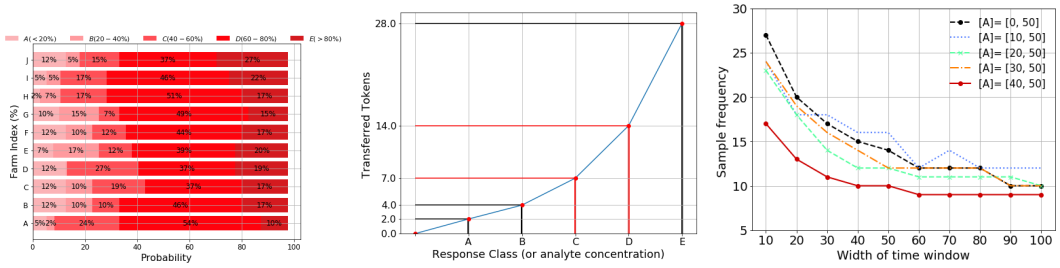
The proposed BC-IoNT system, illustrated in Figure 3.8, consists of three components; IoNT sensors, IoT devices (transaction nodes-TN) and functional nodes (FN). All the devices will facilitate a set of farms to operate collectively in order to make decisions that can subsequently be used to optimize farm management practices effectively. While IoNT sensors monitor data at molecular-scale, the TNs will collect the information and apply initial pre-processing to reduce the data volume before transmitting it to the FNs. FNs are authorized entities (e.g., regional agri-authorities) that execute a smart contract and perform block mining. A smart contract is a set of rules written for executing certain decision actions when a specific set of conditions are met. The smart contract used here is a ML model formed by integrating the Langmuir molecular binding model [95] with the sequential Bayesian updating model. The Langmuir model is used to extract valuable insights from the IoNT sensor data. The sequential Bayesian updating model is used to derive a probability

distribution of the information derived through the Langmuir model, and this will contribute towards the decision making process. Moreover, the space required for storing historical data is reduced since it uses the prior probability distribution that represents historical data. This is a great advantage because the space required for storing BC ledger is a critical issue since the BC systems stores all information in the ledger. Moreover, the proposed approach also incorporates a credit system to determine the credibility of decisions. Finally, the FN creates two blocks as the system maintains two ledgers for the TN and FN networks, and this will be explained further in the subsequent sections.

Furthermore, to prevent any unauthorized access of data communicated between the TNs and the FNs as well as the data stored in the BC system, data is encrypted by using the Advanced Encryption Standard Galois Mode (AES-GCM) method [96]. The AES-GCM is a symmetric encryption method that enables encryption and decryption of data using the same key, which is generated by using the Diffie-Hellman (D-H) key exchange service [97]. The D-H service allows sharing a common secret key (SK) between two or more parties without sending their private keys. The AES-GCM encryption method also generates compressed data in plain-text which contains a key value used for integrity protection and authentication during the decryption phase. Hence, any FN that contains the encrypted key can verify the integrity of the encrypted data and perform an authenticated decryption. Before any TN passes its data to a selected FN, the TN's private key and FN's public key are fed to the D-H method to generate a SK that is used to encrypt the data. TN then sends the encrypted data to the FN and this includes the TN's public key. The FN generates the SK by using its private key and the public key of TN to decrypt the data. The FNs follow the same encryption procedure when sending blocks back to the TN network. Since the private keys stay within the devices where they are generated, this data encryption approach prevents the risk of accessing data by an unauthorized party, thereby protecting data privacy as well as integrity.

In this task, the PhD research demonstrates the use of the proposed system with an application of detecting the level of chemicals used on the farms. This is a critical application for SF that aims to increase sustainable farm productivity coupled with controlled delivery of chemicals such as synthetic fertilizers and herbicides in order to maintain optimal soil quality and minimize the environmental and economic cost. This will also have benefits from the consumer point of view, who are always concerned with food contamination that results from excessive use of chemicals.

The IoNT sensors used here are the affinity-based nanosensors, which can detect the range of chemicals used on farms at the molecular-level (Figure 3.8). The IoT device in each farm collects and then categorizes the IoNT sensor data into five different classes, A to E.



(a) (b) (c)

Fig. 3.9 Selected outcomes of the BC-IoNT system in detecting the level of chemicals on farmlands; (a) Color token for 10 farms, (b) Amount of credits rewarded or penalized with RCs, (c) Variability in the number of samples required for detecting the RC having $p \leq 10^{-3}$ accuracy with time window sizes for different chemical concentrations (Appendix E, (a)- Fig. 7(b) in page 13, (b)- Fig. 10(a) in page 13, and (c)- Fig. 5 in page 11).

Each of these class corresponds to specified ranges of chemical concentrations named as response classes (RCs). For a time period, the frequency of each farm being in the five RCs are computed during the pre-processing phase, and then sent to the FN where block mining is carried out. The smart contract is then executed to compute the probability distribution of the level of a chemicals in each farm belonging to one of the five RCs categories. The probability distribution of each farm is represented as a color token as illustrated in Figure 3.9a, and this is then used to decide the RC that each farm belongs to (the RC which has $\geq 80\%$ probability). The study on detecting the RC of a set of farms proves that the proposed BC-IoNT system improves the performance compared to the centralized approach.

In addition, given a particular concentration range as the optimal range that every farm should maintain (i.e., an optimal RC), the system then decides whether a selected farm is compliant with the optimal RC. Accordingly, each farm is rewarded or penalized a certain amount of credits either belonging or not part of the optimal RC. The amount of credits being rewarded or penalized to a farm is exponentially increased based on the frequency of its compliance with the optimal chemical standard (see Figure 3.9b). This means that the more a farm follows the optimal chemical standard, the more rewarded credits it will obtain. Therefore, the amount of credits held by each farm is an indicator of its credibility of being compliant with the optimal level of chemicals. Thereafter, two blocks are created for the TN and FN networks. The block for the FN network contains all information (i.e., color token and credit transaction of all farms) which is required for performing future block mining, while the block for the TN only needs the color token and the available credits in each farm. Finally, these two blocks are securely shared over the BC networks. Therefore, every FN can

see the variability in the level of a chemical in all farms and each TN can see their current level of chemicals and amount of credits.

This study also examines the optimal data sample frequency that the system requires for detecting the precise levels of chemical and this includes the impact of several parameters such as the variability in field conditions, which can impact on the accuracy of detecting the level of chemicals. For instance, Figure 3.9c depicts variability in the optimal number of samples during a time window for different levels of a particular type of chemical. We explore the variability in farm credits over a period of time for a number of farms and we could easily detect the farms which are compliant with the optimal chemical level. Therefore, the agri-authorities (i.e., FNs) can alert farms which do not maintain the optimal level of a chemicals. On the other hand, they can also share this data with other stakeholders in the food supply chain such as the consumers to check the quality of food that they buy based on the use of chemicals during the production process. For example, in the market, there is a growing trend in buying organic food and this system would be a good fit to facilitate appropriate food selection.

RQ3 - Task 5: Energy Management in Distributed Data Analytics

Although fully distributed computing platforms have the potential to perform data analytics for making decisions effectively, determining the energy required for executing analytical tasks is challenging, and specially in resource-constrained settings. The SF sector is mostly based on battery-powered wireless sensor devices that have limited energy to accomplish different tasks such as sensing, transferring, storing and processing. Therefore, this PhD research proposes **cooperative computing process via computation offloading** in a distributed computing environment for optimizing the use of in-network computational resources (Appendix F). In some devices, particular computational tasks can be executed in parallel. Such tasks are partially offloaded to neighboring devices that have sufficient resources to complete them within a given deadline to optimally utilize available resources within the system. The devices which offload computations partially to neighboring devices are called the Initiation Node (IN) and neighboring devices are named as the cooperative Node (CN).

This task first explores the computational and communication cost associated with device-to-device communications and computations. Second, this energy cost for a cooperative computing environment is modeled, including the micro-solar energy harvesting capacity of sensor devices. Third, this study discusses optimum task partitioning to minimize the total energy consumption based on the energy harvesting status of the sensor nodes for

different scenarios. Moreover, this task addresses four different scenarios for how IN and CN devices manage harvested energy for performing energy-aware cooperative tasks, namely, shadow-shadow, shadow-light, light-shadow, and light-light. This cooperative computing allows for enhancements of the computational capability and delivery of timely analytic outcomes for time-sensitive applications, while also reducing the total energy cost. This energy balance is a vital factor for minimizing the energy waste and long-term operation of devices with minimal maintenance. The Lagrange Multiplier optimization method is used for solving these constrained energy optimization problems. Selecting CN is also a challenging task in a massively distributed environment like SF. Therefore, this research also explores the selection of CNs based on energy-aware and utility-based factors aiming to minimize their over-utilization, while balancing the fairness of selection. The proposed approaches are evaluated using the SmartGrid simulator and the outcomes exhibit reduced energy consumption of the distributed data analytic network along with improved effectiveness of the operational tasks of sensor devices.

3.2 Validation

In order to validate the solutions proposed under the five tasks to address our research questions, this PhD research conducts data analysis using both real and simulated datasets. The application of these datasets under each task is listed below. This research study mainly uses Python tools for the data analysis tasks.

1. A MIRS dataset is used for exploring the performance of the CL and FL-NNPLS techniques proposed in Task 1 and Task 3, respectively. This dataset was collected at the TEAGASC research dairy farm, Moorepark, Ireland, and contains the MIR spectra of 712 different milk samples in the wavenumber region 925-5005 cm^{-1} (1060 wave numbers), along with the composition of several milk quality traits (MQTs) that are used to asses the quality of milk based on their composition in milk.

   The experiments conducted to validate the CL approach examine the optimal compression level that can achieve a similar performance and meaning to the original data. In order to achieve this, the data compression rate is varied while assessing the predictive performance of a set of selected MQTs. Subsequently, the optimal compression parameters that are required to be transferred to the location where the data is processed or stored are selected. Moreover, to emphasize the significance of preserving the original characteristics of the dataset in the CL process, the CL performance is

computed by preserving non-linearity in the MIRS data and then compared it to the CL performance obtained without preserving non-linearity during the CL process. The experiment conducted for validating the FL-NNPLS approach initially examine the predictive performance of the proposed NNPLS model along with the LSQR and PLSR models for a set of MQTs under the CML settings. Following this, the MIRS dataset is equally divided into a few subsets (i.e., clients) to evaluate FL-NNPLS predictive performance for three selected MQTs. Moreover, the predictive performance of the CML and FL-NNPLS methods with Deep Learning models is also compared to justify their performance compared to state-of-the-art algorithms.

2. In the study of deriving novel metrics under Task 2, a Global Positioning System (GPS) mobility dataset of a dairy herd of 30 cows collected for five consecutive days at the same location as the MIRS dataset (location of each cow was recorded every 4 minutes) is used. The GPS data is used to compare the performance of the proposed *AIm* metric in computing the cohesion of dairy cows with the two already existing SNA measures. In doing so, this study examines the variability in the cohesion of 33 cows computed from these three metrics. Moreover, simulated mobility data is also used to demonstrate an application of the insights extracted using the *AIm* metric for characterizing social behaviors in dairy herds. The *Random Wayponit* model developed in the *Pymobility* [98] python package is used to generate a simulated mobility dataset. The Random Waypoint model is commonly used to simulate mobility data based on the velocity, mobility region, and waiting time parameters at a point before making the next movement. This study also explores how the information from the behaviors of sick and estrus cows can be extracted from large-scale complex mobility data by using the *AIm* metric.

3. In Task 4, this PhD research investigates how the BC enabled distributed learning framework can collectively monitor the level of chemicals in farmlands based on simulated chemical level data. The Gaussian and Uniform probability distributions based random number generation Python packages are used to simulate the level of chemicals. This task then experiments on how accurately the BC-IoNT system can detect the level of chemicals over a set of farms compared to the standard CML-based approach, including the impact of several factors such as the sampling frequency in detecting the level of chemicals. The experiment is extended further to study the variability in the credibility of farms in order to determine if they are compliant with the chemical standards during the production process.

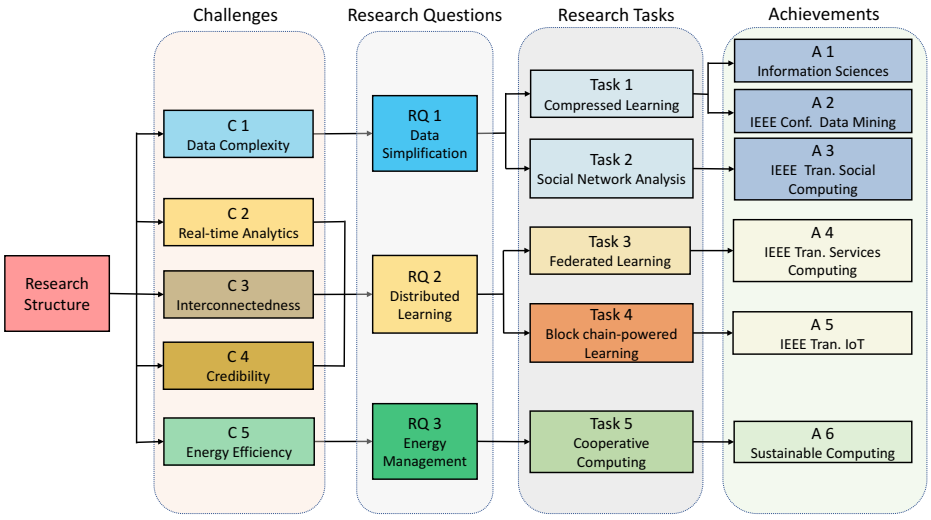


Fig. 3.10 Research Structure and key achievements.

4. In the cooperative computing via computation offloading for energy management study undertaken in Task 5, a prototype-based analysis is performed by using the SimGrid simulator [99]. Simgrid is an open-source simulator that contains packages to explore the behavior of large-scale distributed analytical platforms. An analysis is conducted to explain how the proposed approach can improve overall energy utilization, while minimizing the energy losses and waste.

3.3 The Contribution

This PhD research considers distributed data processing and learning in the context of smart dairy farming. The thesis shows that the five research tasks have made significant contribution towards advancing the future distributed learning systems. This section summarizes the key achievements of the research, while Figure 3.10 illustrates how current challenges in the SF data analytics and the research questions align with the achievement in each task. The achievements 1, 2, and 3 (**A1**, **A2**, and **A3**) are findings of the study carried out under the first research question (RQ1) for developing data processing techniques to enable effective learning from complex and large data set. The **A1** and **A2** relate to the CL for dealing with simplifying large-scale complex data for efficient mining and **A3** proposes a metric that enables extracting insights from complex data. As part of the RQ2, **A4** presents a distributed learning framework integrated with a hybrid ML model (FL-NNPLS) which facilitates performing real-time data analytics by incorporating distributed data. Since the FL has limitations in application under some circumstances, a fully distributed analytical

framework is developed in association with BC technology as the 5^{th} achievement (**A5**). Finally, **A6** emanated from the study conducted under RQ3 and discusses the optimization of energy for performing cooperative data analytics in a distributed environment. Detailed information about the achievements under the corresponding research questions is provided below.

- **RQ1: Data Simplification**
  - **A1:** This PhD research proposed a CL method to mitigate the complexity in large-scale data processing and learning through data compression (Appendix A). CL allows learning to be performed on the compressed data, and thereby avoids restoring the original data complexity during the learning process. Moreover, CL contributes to optimizing resource cost, and facilitates the storage of only the compression parameters and recovery algorithms for long-term data storage. Our analysis shows that the MIRS dataset can be compressed by 90% without impacting the predictive accuracy for a selection of MQTs. Thus, performing CL in proximity to the data source facilitates the performance of data analytics on small datasets as well as reducing data processing, communication and storage overheads in big data analytics. Therefore, CL is a promising alternative for data-intensive, time-sensitive and resource-constrained data analytics applications.
  - **A2:** This is an extension of **A1** and emphasizes the significance of exploring domain-specific knowledge for improving the CL performance (Appendix B). The MIRS data analysis shows that preserving non-linear characteristics in the MIRS dataset improves the CL predictive performance of certain MQTs. Therefore, this highlights the fact that acquiring detailed knowledge on the characteristics of data is essential in selecting proper compression and learning techniques for improving CL performances. Specifically, the way in which CL is executed is critically important. This requires that both the compression and learning take place within the same device. This will ensure that preserving the original characteristics of data during the compression is an application-specific requirement. On the other hand, if the compression and learning takes place at two different locations, then preserving the original characteristics of data is very important as there is no information about the learning tasks that will be carried out. Therefore, the use of domain-specific knowledge in CL minimizes the loss in the original characteristics of data in the compression phase and avoids the unnecessary use of heavy computational techniques, while preserving the CL performances.

  - **A3:** This thesis proposed a novel metric in combination with other existing metrics to demonstrate an application for explaining underlying dynamics in large-scale complex data (Appendix C). The metrics are social interaction range and social affiliation matrix, and when combined can transform complex and heterogeneous social interactions into a social network graph that can express the most significant social interactions. This simplifies the presentation of complex social interaction data, which in turn improves the representation of social interactions using a network graph. The *AIm* metric then incorporates network (group)-level information for characterizing node (individual)-level behaviors. This expands the range of information taken into account for characterizing individual behavior compared to the already existing measures that use intermediate-level information only. The data analysis conducted for performance evaluation confirms that the *AIm* metric has greater stability in computing cohesion of animals herd compared to the already existing SNA measures. Besides, the study for identifying sick and estrus animals by using the *AIm* metric emphasizes that atypical social behaviors can be identified effectively by exploring the evolution of *AIm* at node-level as well as group-level. Therefore, *AIm* metric along with social interaction range and social affiliation matrix simplify a large amount of social behavioral data and enable effective extraction of valuable insights into the social relationships which can be used to determine the health conditions of the cows.

- **RQ2: Distributed Learning**

  - **A4:** The present study proposed a FL-based distributed learning framework combined with a hybrid NNPLS model, and this is known as FL-NNPLS (Appendix D). While the NNPLS model overcomes several issues that limit the use of current ML models, the FL enables collaborative training of the NNPLS model by integrating distributed data sources without moving data out from data sources. This framework realizes real-time data analytics, alleviating critical challenges that found in numerous DML models such as data ownership, privacy and security and optimizing resource utilization effectively. The FL-NNPLS performance analysis confirms that the framework can achieve comparable performance to the CML approach. Moreover, the re-sampling and zero padding based joint approach mitigates the data imbalance issue in DML. Therefore, the FL-NNPLS data analytics has strong potential to address issues that lie in environments containing privately-held data.

- **A5:** As an extension of **A4**, this PhD research proposed a fully distributed analytic framework by integrating IoNT with the BC technology (BC-IoNT) (Appendix E). This is a new direction in distributed data analytics systems since there has been no attempt in combining the IoNT paradigm to BC-based systems. This combined system enables monitoring of data at the molecular-level and enhances the potential of operating fully distributed data analytics for decision-making, allowing direct interaction between distributed data sources that was not possible in **A4**. The contribution in this thesis lies in compliance levels of chemical usage on farms. The thesis proposes a ML model formed by joining the Langmuir molecular binding model with the sequential Bayesian updating method to detect the level of the chemicals, and the model outcome is categorized into color token. In addition, a token-based credit transaction mechanism is introduced to quantify the credibility of farms being compliant with chemical standards. The system shares the color token and the amount of credit that each farm holds over the BC network to ensure the transparency, traceability and credibility of farms in order to determine if they are compliant with the chemical standards. These are the essential features that are lacking in current food production and supply chains. Therefore, the BC-IoNT system can provide opportunities not only for employing sustainable farming practices but also for managing an efficient supply chain. Our simulation experiments showed that the BC-IoNT system has greater performance compared to the centralized approach in detecting the level of chemicals. Specifically, the accuracy of the BC-IoNT was $\geq 90\%$ and the centralized approach was $\leq 80\%$. The study found that the efficiency of detecting the level of chemicals depends on the sampling frequency and the variability in chemical level on farms.

- **RQ3: Energy management**

  - **A6:** This thesis proposed a theoretical framework for cooperative computing in distributed data analytics environment via computation offloading for micro-solar powered heterogeneous WSN (Appendix F). This research discussed optimum data partitioning to minimize the total energy consumption in computation and communication based on the energy harvesting status of sensor nodes for different scenarios. Based on evaluations conducted using the theoretical models proposed in this study, the results show that there is a reduction of both the energy losses and waste in response to energy conversion and overflows compared to a

data partitioning algorithm that offloads computation tasks without taking the energy harvesting status of nodes into consideration. The proposed approach also improves the energy balance of distributed sensor devices for long-term sustainable operation. Aiming to minimize the over-utilization of devices, this study also considered energy-aware node selection for executing cooperative computing based on a utility function.

As part of this research, these achievements have been published as research articles. They are the outcomes of the research tasks that are carried under three RQs (see Figure 3.10) and the output consist of five journal articles and one conference paper; four of them are accepted and the remaining two (**A4** and **A5**) papers are under review. The articles are listed below and their original copies are included in the Appendix *A* to *F*.

A1. D. Vimalajeewa, C. Kulatunga, and D. P. Berry, ***Learning in the compressed data domain: Application to milk quality prediction*** , Information Sciences, vol. 459, no. 2, pp. 149-167, May. 2018 (Appendix A).

A2. D. Vimalajeewa, E. Robson, D. P. Berry, and C. Kulatunga, ***Evaluation of Non-linearity in MIR Spectroscopic data for Compressed Learning***, High Dimensional Data Mining (HDM) Workshop, IEEE Conference on Data Mining , New Orleans, USA, (ICDM 2017), pp. 545-553, Nov. 2017 (Appendix B).

A3. D. Vimalajeewa, S. Balasubramaniam, B. O'Brian, and D. P. Berry, ***Leveraging Social Network Analysis for Evaluating Animal Cohesion***, IEEE Transactions on Computational Social Systems, vol. 6, no. 2, pp. 323-337, Mar. 2019 (Appendix C).

A4. D. Vimalajeewa, C. Kulatunga, D. P. Berry, and S. Balasubramaniam, ***A Service-based Joint Model Used for Distributed Learning: Application for Smart Agriculture***, submitted to IEEE Transactions on Services Computing, (**under review, submitted July 2019**) (Appendix D).

A5. D. Vimalajeewa, S. Thakur, J. Breslin, D. P. Berry, and S. Balasubramaniam, ***Blockchain-Powered IoT system integrated with IoNT for Smart Farming***, submitted to IEEE Transactions on Internet of Things, (**under review, submitted Nov. 2019**) (Appendix E).

A6. C. Kulatunga, K. Bhargava, D. Vimalajeewa, and S. Ivanov, ***Cooperative in-network computation in energy harvesting device clouds***, Sustainable Computing: Informatics and Systems, vol. 16, pp 106-116, Dec. 2017 (Appendix F).

3.4 Answers for the Research Questions

This section presents the answers to the research questions described in section 1.2.2.

- **RQ1-Simplifying Data:** *What are the optimal techniques and metrics for simplifying and explaining complex data with minimal loss of information?*
  This research has shown that the CL approach enables applying data analytics on the compressed version of the original data and can achieve similar learning performance to cases when only the original data were used (**A1**- Appendix A). The study also found that the incorporation of domain-specific knowledge into the CL process also enhances the CL performances, and most importantly, enables selecting the best fit compression and learning methods, preventing the unnecessary use of resource intensive methods such as DL (**A2**-Appendix B). Since CL allows the use of a compressed version of the original data for applying data analytics and it avoids the need for decompression of data prior to applying data analytics unlike conventional data compression approaches. Thereby, it removes the need to restore the original data complexity in the learning stage, allowing the data analytics to be conducted effectively. At the same time, storing only the optimal compression parameters along with recovery algorithms helps to reduce the need for transferring large volumes of data and the space required for long-term data storage. The research has also proved that the *AIm* metric along with the social interaction range and the social affiliation matrix can simplify complex social behavioral data to characterize underlying dynamics effectively (**A3**-Appendix C). Moreover, exploring the variability in *AIm* can be used in SF applications for detecting atypical social behaviors which can be used to make early warnings when the animal's health condition has deteriorated.

  Therefore, the CL approach and the *AIm* metric can be capitalized to extract the most meaningful information from large-scale complex datasets, thereby supporting efficient data processing, transferring and storing. These characteristics makes the CL approach an ideal solutions in SF applications, which in particular have limited resources for processing large volumes of complex data.

- **RQ2-Distributed Learning:** *What frameworks can be designed to make data analytics effective by securely incorporating distributed data sources?*

  The PhD research has proposed two distributed data analytics approaches, which are FL-NNPLS (**A4**-Appendix D) and BC-IoNT (**A5**-Appendix E) to enable timely and accurate analysis of data using distributed data sources.

  The FL-NNPLS approach allows the combined use of NN and PLSR models to overcome the drawbacks that limit their application in data analytics when the algorithms are utilized on their own, and enables cooperative training for the joint NNPLS model in real-time manner for distributed data sources. The performance evaluation showed that the join model can achieve similar learning performances as the standard CML approach, mitigating several limitations in distributed data analytic systems such as data sharing, ownership, and privacy found in typical DML frameworks. However, The FL-NNPLS framework relies highly on the central service unit where the final updated model is refined and updated. This means that the stability as well as the credibility of the learning process can mainly be affected due to functional failure or misbehavior of the central service unit. The approach also lacks a mechanism to validate the quality of data that the clients will use for training the model and also the validity of their local model updates. The PhD research found that the limitation on the validity of the data and preservation of data privacy can be maintained when the BC-IoNT system is incorporated. This integration facilitates secure storage of valid information and subsequent sharing of it over distributed clients. This also ensures the traceability and credibility of the information stored in the ledger as well as the decisions made by the system. By incorporating nanosensors through the IoNT paradigm, the BC-IoNT system allows monitoring of data at molecular-level, and the Langmuir molecular binding model combined with the Bayesian theory-based smart contract is used for data processing and decision-making. The research demonstrated the use of this system for detecting the level of chemicals in farmlands and found that it can detect the level of chemicals with higher accuracy compared to the centralized data analytics approach.

  Therefore, these two learning frameworks are promising solutions for performing distributed data analytics for systems with distributed data sources. These learning frameworks will be ideal for SF, where owners can be comfortable with data located locally in private repositories as farmers and stakeholders are reluctant to share their data.

- **RQ3-Energy Management:** *How can energy be flexibly managed for systems that require distributed data analytics?*
  The PhD research has considered performing cooperative data analytics by partitioning tasks and offloading them to neighboring devices for optimizing computational and communication resources (**A6**-Appendix F). The theoretical models developed for solar energy harvesting in WSN shows that the computation offloading to peer nodes that have sufficient resources, can significantly reduce the total energy consumption with improved energy balance across the network. Besides, the data simplification techniques and distributed computing frameworks proposed in the present study can also effectively reduce the energy consumption in distributed data processing and analytics. This means that the CL and the FL-NNPLS framework can minimize the need for transferring large volumes of data over the distributed network, which in turn, saves the energy required for computing and communication and increases the data processing and analytic efficiency.

As illustrated in Figure 3.10, each of these achievements addresses the challenges associated with current data processing and learning at different stages in the SF process. They are, however, not limited to SF only and can be applied to diverse application domains. The achievements resulting from RQ1 (**A1**, **A2**, and **A3**) lays the foundation for preparing the data that can be processed during the learning process, mitigating the challenges that arise due to complex and voluminous data (**C1**). The CL enables performing learning through small and meaningful data which is particularly beneficial in applications where there is limited computing, communication, and storage resources. Moreover, the *AIm* metric is a novel concept that emphasizes the significance of using SNA in the SF sector and creates opportunities for extracting valuable information from large-scale complex social behavioral data. Another milestone of this research is the distributed learning frameworks (**A4** and **A5**) that allows data analytics to be conducted over distributed environments, mitigating the challenges mentioned in **C2** as well as **C3**. In particular, the BC-powered IoT system integrated with IoNT (**A5**) unlocks the opportunities for practicing peer-to-peer interactions with improved guarantee of data security, transparency and auditability of information traversing across a distributed learning platform (**C4**). Finally, the last key milestone of this research is obtained via addressing **C5**. The research designed an analytical framework for cooperative computing via computation offloading aiming to optimize the energy consumption in a distributed environment (**A6**).

3.5 Summary

This chapter briefly discussed the research conducted to address the three research questions defined in chapter 1; (1) simplifying large-scale complex data, (2) performing distributed data analytics and (3) resource management of low-powered communication networks in distributed data analytics. The present PhD research mainly focused on developing novel data analytics techniques using state-of-the-art machine learning methods and utilizing them to analyze data in the smart dairy farming environments in order to address the research questions.

Firstly, this study addressed the RQs under five research tasks and proposed novel data analytic techniques. These techniques include Compressed Learning (CL) and Animal Importance (*AIm*) metric to minimize the complexity for processing large-scale complex data, a hybrid ML model integrated with the federated learning (FL-NNPLS) system and Block chain integrated with IoNT (BC-IoNT) system for performing distributed learning by incorporating distributed data sources. This is followed by a cooperative computing via computation offloading method to optimally utilize the available energy in distributed data analytics. Secondly, the PhD research discussed the tools and data analysis used to validate the proposed techniques. The research considered both real and simulated datasets to validate the proposed algorithms and models to prove that they have state-of-the-art performance. Thirdly, the proposed solutions are briefly discussed as research achievements that has resulted in six research articles. Finally, the chapter maps the achievements to address each of the research questions.

Chapter 4

Conclusion and Future Work

This chapter presents the conclusion in section 4.1 that focuses on addressing the three research questions, and this is followed by the future work in section 4.2.

4.1 Conclusion

The advances in modern IoT and ICT technologies has resulted in new technologies that can be applied to SF. One specific element in SF is the collection of data that are produced and transmitted from these technologies. However, as we increase the number of these sensing devices on the farm, the collected data becomes high and can be complex. These data have to be processed, transformed and stored, utilizing available resources effectively to make decisions in order to optimize the SF management process. The primary goal of this research was, therefore, to address some key challenges in distributed data processing and learning using state-of-the-art computational techniques in the context of smart dairy farming. The research formulated those challenges into three research questions and then addressed them under five tasks to propose solutions.

Initially, the thesis explores effective data compression techniques that can be used to simplify large-scale complex data in order to perform data analytics effectively. Based on this, the thesis proposes the compressed learning method that allows learning from compressed data. By compressing the data, this can minimize the cost of transmission. Most importantly the results from the compressed learning resulted in minimum performance loss compared to learning from using only the original data. The thesis also considers the challenge from high complexity nature of the collected data that is limited from using existing metrics that defines underlying dynamics of data. Based on this, the thesis proposes a novel metric known as *Animal importance (AIm)*, which is used for understanding the

dynamics in large-scale complex data emanating from animal group behaviour. Secondly, the thesis derived novel ML models and performed distributed data analytics considering the limitations of existing ML models in distributed environments. The research proposed a hybrid ML model combining the ML models NN and PLSR (NNPLS) to mitigate the drawbacks that limit their individual use in data analytics. Subsequently, a FL-based DML framework (FL-NNPLS) was used to train the hybrid model for performing distributed data analytics. Considering the limitations in the FL-NNPLS framework along with other prevailing concerns in existing DML systems such as direct interactivity, traceability and transparency, the research proposed a fully distributed learning framework by integrating the BC technology with IoNT (BC-IoNT). Finally, the thesis presents a solution for optimizing the available energy in distributed computing systems based on energy-aware cooperative data analytics via computation offloading. Each of these solutions will advance SF by enabling various factors to be considered, ranging from heterogeneous and scale of sensors, including their computational capabilities, as well as techniques to efficiently manage privacy of data in a distributed farm environments. Each of these solutions will result in new generation and novel ICT technologies that will result in efficient, cooperative and intelligent distributed data processing and learning platforms.

4.2 Future Work

The data processing and learning methods proposed in this PhD research will open several future research directions to further improve SF performance as well as flexibility to address any changes that cannot be foreseen in the future. This section lists a number of potential research directions that can be taken forward into the future.

- **Enhancing the performance of existing systems:** In distributed computing systems, the selection of data processing nodes is vital in terms of improving decision-making efficiency as well as the optimal utilization of available resources. The *AIm* metric that was used to define the condition of the animals can be used to identify the most and least connected devices in a distributed computing network. Subsequently, devices with high *AIm* values can be used as the coordination device in the FL-NNPLS framework to improve the effectiveness of data communication in the network in order to minimize energy cost from transmission. This is because their greater connectivity with peers can assist in communication options that can result in network efficiency, and in particular for wireless systems. Meanwhile, devices that have low *AIm* are suitable for offloading

computations, sharing resource requirements, and for mining nodes in the BC-IoNT system. Mining is an important function in BC and requires devices with very high computational capabilities. Thus, incorporating the *AIm* metric and the cooperative computation offloading method with the FL-NNPLS and BC-IoNT systems can further improve the overall performance of the system. An example improvement is the NNPLS model updating speed and block propagation speed over the BC networks in the FL-NNPLS and BC-IoNT systems. Therefore, this is an interesting direction to continue this research to enhance the effectiveness of the FL-NNPLS and BC-IoNT.

In addition, the effectiveness of the BC-IoNT can be enhanced further by strengthening the effectiveness of the smart contract in detecting the level of chemicals. The smart contract used in the BC-IoNT system considered only the dynamics of molecules of a certain chemical in a medium. However, in reality, there can be other chemicals such as contaminants which can have an impact on the functionality of the affinity-based nanosensors. Hence, the accuracy of the smart contract's outcomes can be enhanced further by considering the dynamics of these factors (e.g., disruption of contaminants) in deriving the ML model to determine the level of chemicals. Moreover, integrating the proposed BC-IoNT system with the existing BC networks (i.e., building consortium BC) can also be an interesting extension of this study. This can facilitate various stakeholders to join the BC network and also expand the range of parameters that BC can be taken into account to develop an efficient food supply chain. For example, developing advanced decision-making systems by integrating with BC-based systems such as Nori and Regen. While *Nori* [67] was developed for reversing the climate change through reducing CO_2 emission, *Regen Network* [68], the mechanism can be further extended to monitor ecological degradation and climate change. Integrating these two technologies with the BC-IoNT system can bring factors of climate change into new sustainable practices for future agri-food production and supply chain.

- **Exploration of the inter-relationships:** Introducing more meaningful variables into the ML models proposed in this study and then making them adaptive to time-variant dynamics is essential for producing meaningful and timely accurate insights for decision-making. This is essential to expand the range of dynamics that they can take into account and then make more context-sensitive decisions. Exploring inter-relationships between diverse parameters and formulating mathematical and statistical approaches to represent their relationships are essential in future research to improve the capabilities of all the proposed models investigated in this thesis. For example, in the context of

milk quality analysis, incorporating variables such as systematic environmental effects such as prevailing weather factors, feed intake, and herd dynamics into the milk quality predictive models can enhance the capability of producing more meaningful decisions for better farm management. At the same time, incorporating information coming from different stakeholders such as consumer buying behavior and market demand into the process of quantifying the credibility of farms being compliant with chemical standards.

- **Autonomous analytic systems:** WSN are increasingly being proposed for various SF applications. In the future, the design of more powerful and energy-harvesting capable sensors along with novel communication systems will facilitate increased data analytics at the edge of the network. This will enhance the capability of real-time analytics and also controlling or adjusting the entire network autonomously. The set of ML techniques proposed in this study has strong potential to perform data analytics by incorporating such devices with low-energy footprint. However, further works such as exploring computational complexity, flexibility, and accuracy based on different objective functions are essential to make them essential for WNS-based SF environments.

Bibliography

[1] O. Elijah, T. A. Rahman, I. Orikumhi, C. Y. Leow, and M. N. Hindia. An overview of internet of things (iot) and data analytics in agriculture: Benefits and challenges. *IEEE Internet of Things Journal*, 5(5):3758–3773, Oct 2018.

[2] N. Ramankutty, Z. Mehrabi, K. Waha, L. Jarvis, C. Kremen, M. Herrero, and L. H. Rieseberg. Trends in global agricultural land use: Implications for environmental health and food security. *The Annual Review of Plant Biology*, 69(14):1–27, 2018.

[3] H. Sundmaeker, C. Verdouw, S. Wolfert, and L. Perez-Freire. *Internet of Food and Farm 2020*, chapter 4. River, 2016.

[4] V. Cabrera. *DairyMGT: A Suite of Decision Support Systems in Dairy Farm Management*, chapter 1. University of Wisconsin-Madison, 2012.

[5] Z. Lu, X. Sun, and T. La Porta. Nanotechnologies in agriculture: New tools for sustainable development. *Trends in Food Science & Technology*, 22:585–594, Dec 2011.

[6] J. Qiu, Q. Wu, G. Ding, Y. Xu, and Shuo Feng. A survey of machine learning for big data processing. *EURASIP Journal on Advances in Signal Processing*, 1:67, 2016.

[7] M. J. O'Grady and G. M. P. O'Hare. Modelling the smart farm. *Information Processing in Agriculture*, 4:179–187, 2017.

[8] J. Konecny, H. B. McMahan, D. Ramage, and P. Richtarik. Federated optimization: Distributed machine learning for on-device intelligence. *arxiv*, 2016.

[9] C. Tsai, C. Lai, M. Chiang, and L. T. Yang. Data mining for internet of things: A survey. *IEEE Communications Surveys Tutorials*, 16(1):77–97, 2014.

[10] V. Sucharitha, P. Prakash, and G. N. Iyer. Agrifog- a fog computing based iot for smart agriculture. *International Journal of Recent Technology and Engineering (IJRTE)*, 7(6):2277–3878, 2019.

[11] Apache. Apache storm. http://hortonworks.com/hadoop/storm/, 2011. [Online; accessed Sep. 2018].

[12] Apache. The apache spark project. http://spark.apache.org/, 2012. [Online; accessed Aug. 2016].

[13] J. Dean and S. Ghemawat. Mapreduce: Simplified data processing on large clusters. *ACM Communication*, 51(1):107–113, 2008.

[14] Horovod. Distributed training framework for tensorflow, keras, pytorch, and apache mxnet. https://github.com/horovod/horovod, 2019. [Online; accessed Oct. 2019].

[15] V. K. Potluru, J. Diaz-Montes, A. D. Sarwate, S. M. Plis, V. D. Calhoun, B. A. Pearlmutter, and M. Parashar. Cometcloudcare (c3): Distributed machine learning platform-as-a-service with privacy preservation. 2014.

[16] C. Wu, J. Xiao, G. Huang, and F. Wu. Galaxy learning. *arXiv*, 2019.

[17] E. P. Xing, Q. Ho, W. Dai, J. K. Kim, J. Wei, S. Lee, X. Zheng, P. Xie, A. Kumar, and Y. Yu. Petuum: A new platform for distributed machine learning on big data. *IEEE Transactions on Big Data*, 1(2):49–67, 2015.

[18] Food and Agriculture Organizations (FAO). *The future of food and agriculture – Trends and challenges*. The United Nations, Rome, 2017.

[19] K. Bhargava, S. Ivanov, and W. Donnelly. Internet of nano things for dairy farming. In *Proceedings of the Second Annual International Conference on Nanoscale Computing and Communication*, NANOCOM' 15, pages 1–2. ACM, 2015.

[20] C. Lokhorst. *An Introduction to Smart Dairy Farming, published*. Van Hall Larenstein University of Applied Sciences, P.O Box 1528, 8901BV Leeuwarden, The Netherlands, 2018.

[21] A. Connolly. Bridging the data gap in dairy farming: The promise of digital technologies. https://www.alltech.com/blog/bridging-data-gap-dairy-farming-promise-digital-technologies, 2018. [accessed Oct. 2019].

[22] Stellapps. One-stop dairy supply chain digitization via iot. http://www.stellapps.com/, 2017. [accessed Aug. 2019].

[23] AgriWebb. Agriwebb: Farm and livestock management software. https://www.agriwebb.com/, 2017. [accessed Aug. 2019].

[24] KEENAN. Keenan diet feeders, mixer wagons, alltech. https://www.alltech.com/keenan, 2018. [accessed Oct. 2019].

[25] C. J. Rutten, A. G. J. Velthuis, and W. Steeneveld H. Hogeveen. Invited review: sensors to support health management on dairy farms. *Dairy Science*, 96(4):1928–1952, Feb. 2013.

[26] A. Grogan. Smart farming. *Engineering Technology*, 7(6):38–40, July 2012.

[27] C. Lokhorst, R. M. de Mol, and C. Kamphuis. Invited review: Big data in precision dairy farming. *animal*, pages 1–10, 2019.

[28] S. Zhongfu, D. Keming, Z. Feixiang, and Y. Shouyi. Perspectives of research and application of big data on smart agriculture. 2013.

[29] A. Ali, J. Qadir, R. Rasool, A. Sathiaseelan, A. Zwitter, and J. Crowcroft. Big data for development: applications and techniques. *Big Data Analytics*, 1(2), 2016.

[30] Y. Voutos, P. Mylonas, J. Katheniotis, and A. Sofou. A survey on intelligent agricultural information handling methodologies. *Sustainability*, 11:3278, 2017.

[31] J. Rhee, J. Im, and G. J. Carbone. Monitoring agricultural drought for arid and humid regions using multi-sensor remote sensing data. *Remote Sensing of Environment*, 12(15):2875–2887, 2010.

[32] L. Padua, P. Marques, J. Hruska, T. Adao, E. Peres, and J. J. Sousa. Multi-temporal vineyard monitoring through uav-based rgb imagery. *Remote Sensing of Environment*, 10(15):1907, 2018.

[33] G. Bai, Y. Ge, W. Hussain, P. S. Baenziger, and G. Graef. A multi-sensor system for high throughput field phenotyping in soybean and wheat breeding. *Computers and Electronics in Agriculture*, 128:181–192, 2016.

[34] F. Balducci, D. Impedovo, and G. Pirlo. Machine learning applications on agricultural datasets for smart farm enhancement. *Machines*, 6(38), 2018.

[35] P. Murthy, A. Bharadwai, P. A. Subrahmanyam, A. Roy, and S. Rajan. Big data taxonomy. https://cloudsecurityalliance.org/research/big-data/, 2014. [Online; accessed Oct. 2019].

[36] M. Mohammadi, A. Al-Fuqaha, S. Sorour, and M. Guizani. Deep learning for iot big data and streaming analytics: A survey. *IEEE Communications Surveys Tutorials*, 20(4):2923–2960, 2018.

[37] S. Salcedo-Sanz, R. C. Deo, L. Carro-Calvo, and B. Saavedra-Moreno. Monthly prediction of air temperature in australia and new zealand with machine learning algorithms. *Theoretical and Applied Climatology*, 125:13–25, 2016.

[38] Y. Guo, G. Poulton, P. Corke, G. J. Bishop-Hurley, T. Wark, and D.L.Swain. Using accelerometer, high sample rate gps and magnetometer data to develop a cattle movement and behaviour model. *Ecological Modelling*, 220(17):2068–2075, 2009.

[39] K. Hempstalk, S. McParland, and D. P. Berry. Machine learning algorithms for the prediction of conception success to a given insemination in lactating dairy cows. *Dairy Science*, 98(8):5262–5273, 2015.

[40] G. Jaurena, J. M. Cantet, J. I. Arroquy, R. A. Palladino, M. Wawrzkiewicz, and D.Colombatto. Prediction of the ym factor for livestock from on-farm accessible data. *Livestock Science*, 77, 2015.

[41] I. Ali, F. Cawkwell, E. Dwyer, and S. Green. Modeling managed grassland biomass estimation by using multitemporal remote sensing data - a machine learning approach. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(7):3254–3264, July 2017.

[42] R. L. Brotzman, N. B. Cook, K. Nordlund, T. B. Bennett, A. G. Rivas, and D. Dopfer. Cluster analysis of dairy herd improvement data to discover trends in performance characteristics in large upper midwest dairy herds. *Dairy Science*, 98(5):3059–3070, Feb. 2015.

[43] S. M. Patil and R. Sakkaravarthi. Internet of things based smart agriculture system using predictive analytics. In *Asian Journal of Phameceutical and Clinical Scieces*, 2017.

[44] J. Trygg and S. Wold. Pls regression on wavelet compressed nir spectra. *Chemometrics and Intelligent Laboratory Systems*, 42:2019–220, 1998.

[45] M. Jian, K. Lam, and J. Dong. Facial-feature detection and localization based on a hierarchical scheme. *Information Sciences*, 262:1–14, 2014.

[46] R. Hasegawa and K. Hotta. Plsnet: A simple network using partial least square regression for image classification. In *IEEE International Conference on pattern recognition (ICPR)*. IEE, 2016.

[47] R. Tan, G. Xing, B. Liu, J. Wang, and X. Jia. Exploiting data fusion to improve the coverage of wireless sensor networks. *IEEE/ACM Transactions on Networking*, 20(2):450–462, Apr 2012.

[48] Z. J. Zhang, C. F. Lai, and H. C. Chao. A green data transmission mechanism for wireless multimedia sensor networks using information fusion. *IEEE Wireless Communications*, 21(4):14–19, Aug 2014.

[49] C. Teng, K. Brown, C. Caro, W. Nielsen, and J. Wells. A service oriented livestock management system using occasionally connected mobile-cloud architecture. In *2012 IEEE International Systems Conference (SysCon) 2012*, pages 1–5, March 2012.

[50] S. Sivamani, J. Park, C. Shin, K. Cho, D. Park, and Y. Cho. Towards an intelligent livestock farm management using owl-based ontology model. *International Journal of Smart Home*, 9(4):251–266, 2015.

[51] F. J. Ferrandez-Pastor, J. M. Garcia-Chamizo, M. Nieto-Hidalgo, and J. Mora-Martinez. Precision agriculture design method using a distributed computing architecture on internet of things context. *Sensors*, 18(6):1731, 2018.

[52] H. Rahman and R. Rahmani. Enabling distributed intelligence assisted future internet of things controller (fitc). *Applied Computing and Informatics*, 14(1):73 – 87, 2018.

[53] S. Teerapittayanon, B. McDanel, and H. T. Kung. Distributed deep neural networks over the cloud, the edge and end devices. In *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*, pages 328–339, June 2017.

[54] P. Marino, F. P. Fontan, M. A. Dominguez, and S. Otero. Wireless network implementation for viticulture information systems. In *2009 IEEE International Symposium on Industrial Electronics*, pages 932–936, July 2009.

[55] Y. Jiber, H. Harroud, and A. Karmouch. Precision agriculture monitoring framework based on wsn. In *2011 7th International Wireless Communications and Mobile Computing Conference*, pages 2015–2020, July 2011.

[56] G. Sehgal, B. Gupta, K. Paneri, K. Singh, G. Sharma, and G. Shroff. Crop planning using stochastic visual optimization. In *2017 IEEE Visualization in Data Science (VDS)*, pages 47–51, Oct 2017.

[57] C. Baseca, S. Sendra, J. Lloret, and Jesus Tomas. A smart decision system for digital farming. *Agrnomy*, 9:216, 2019.

[58] Decentralized Machine Learning. Decentralized machine learning white paper. https://decentralizedml.com/DML_whitepaper_31Dec_17.pdf, 2017. [Online; accessed Oct. 2019].

[59] T. S. Brisimia, R. Chena, T. Melac, A. Olshevskya, I. C. Paschalidisa, and W. Shi. Federated learning of predictive models from federated electronic health records. *Medical Informatics*, 112:57–69, 2018.

[60] J. Konecny, H. B. McMahan, F. X. Yu, P. Richtrik, A. T. Suresh, and D. Bacon. Federated learning: Strategies for improving communication efficiency. In *IEEE Neural Information Processing Systems Workshop*. IEEE, 2016.

[61] J. Weng, J. Weng, M. Li, Y. Zhang, and W. Luo. Deepchain: Auditable and privacy-preserving deep learning with blockchain-based incentive. *IACR Cryptology ePrint Archive*, 2018:679, 2018.

[62] S. Wang, J. Wang, X. Wang, T. Qiu, Y. Yuan, L. Ouyang, Y. Guo, and F. Wang. Blockchain-powered parallel healthcare systems based on the acp approach. *IEEE Trans. on Computational Social Systems*, 5(4):942–950, 2018.

[63] Z. Yang, K. Yang, L. Lei, K. Zheng, and V. C. M. Leung. Blockchain-based decentralized trust management in vehicular networks. *IEEE Internet of Things*, 6(2):1495–1505, 2019.

[64] J. Kang, R. Yu, X. Huang, S. Maharjan, Y. Zhang, and E. Hossain. Enabling localized peer-to-peer electricity trading among plug-in hybrid electric vehicles using consortium blockchains. *IEEE Trans. on Industrial Informatics*, 13(6):3154–3164, 2017.

[65] M. A. Ferrag, M. Derdour, M. Mukherjee, A. Derhab, L. Maglaras, and H. Janicke. Blockchain technologies for the internet of things: Research issues and challenges. *IEEE Internet of Things*, 6(2):2188–2204, April 2019.

[66] W. Hong, Y. Cai, Z. Yu, and X. Yu. An agri-product traceability system based on iot and blockchain technology. In *2018 1st IEEE International Conference on Hot Information-Centric Networking (HotICN)*, pages 254–255, Aug 2018.

[67] Nori. Remove your carbon footprint. https://nori.com/, 2019. [Online; accessed Oct. 2019].

[68] Ragen Network. Realizing the ecomomics of agriculture. https://www.regen.network/, 2019. [Online; accessed Oct. 2019].

[69] G. Sylvester. *E-Agriculture in Action: Blockchain for Agriculture*. The Food and Agriculture Organization of the United States and the International Telecommunication Union, Bankok, 2019.

[70] AgriDigital. We're building blockchain-based solutions for global agricultural supply chains. https://www.agridigital.io/products/blockchain, 2019. [accessed Aug. 2019].

[71] Xiaoqi Li, Peng Jiang, Ting Chen, Xiapu Luo, and Qiaoyan Wen. A survey on the security of blockchain systems. *arXiv*, 2018.

[72] Ali Dorri, Salil S. Kanhere, , and Raja Jurdak. Blockchain in internet of things: Challenges and solutions. *arxiv.org*, 2016.

[73] A. Destounis, G. S. Paschos, and I. Koutsopoulos. Streaming big data meets backpressure in distributed network computation. In *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, pages 1–9, April 2016.

[74] Z. Sheng, C. Mahapatra, V. C. M. Leung, M. Chen, and P. K. Sahu. Energy efficient cooperative computing in mobile wireless sensor networks. *IEEE Transactions on Cloud Computing*, 6(1):114–126, Jan 2018.

[75] A. Giridhar and P. R. Kumar. Toward a theory of in-network computation in wireless sensor networks. *IEEE Communications Magazine*, 44(4):98–107, April 2006.

[76] Z. Lu, X. Sun, and T. L. Porta. Cooperative data offloading in opportunistic mobile networks. In *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, pages 1–9, April 2016.

[77] C. Shi, V. Lakafosis, M. H. Ammar, and E. W. Zegura. Serendipity: Enabling remote computing among intermittently connected mobile devices. In *Proceedings of the Thirteenth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc'12, pages 145–154, 2012.

[78] N. Dang, E. Bozorgzadeh, and N. Venkatasubramanian. Quares: A quality-aware renewable energy-driven sensing framework. *Sustainable Computing: Informatics and Systems*, 2:171–183, 2012.

[79] G. Zhuo, Q. Jia, L. Guo, M. Li, and P. Li. Privacy-preserving verifiable set operation in big data for cloud-assisted mobile crowdsourcing. *IEEE Internet of Things*, 4(2):572–582, April 2017.

[80] A. Andrawes, R. Nordin, and M. Ismail. Wireless energy harvesting with cooperative relaying under the best relay selection scheme. *Energies*, 12:892, 2019.

[81] M. Amjad, A. Ahmed, M. Naeem, M. Awais, W. Ejaz, and A. Anpalagan. Resource management in energy harvesting cooperative iot network under qos constraints. *Sensors*, 18:3560, 2018.

[82] S. J. C. Janssen, C. H. Porter, A. D. Moore, I. N. Athanasiadis, I. Foster, J. W. Jones, and J M. Antle. Toward a new generation of agricultural system data, models, and knowledge products:state of the agricultural systems science. *Agricultural Systems*, 155:200–212, 2017.

[83] S. Wolfert, L. Ge, C. Verdouw, and M. Bogaardt. Big data in smart farming- a review. *Agricultural Systems*, 153:69–80, 2017.

[84] J. W. Jones, J. M.Antle, B. Basso, K. J. Boote, R. T. Conant, I. Foster, H. C. J. Godfray, M. Herrero, R. E.Howitt, S. Janssen, B. A. Keating, R. M. Chery, H. P. C. Rosenzweig, and T. R. Wheeler. Toward a new generation of agricultural system data, models, and knowledge products:state of the agricultural systems science. *Agricultural Systems*, 155:269–288, 2017.

[85] C. Kempenaar, C. Lokhorst, E. J. B. Bleumer, R. F. Veerkamp, T. Been F. K. van Evert M. J. Boogaardt, L. Ge, J. Wolfert, C. N. Verdouw, M. van Bekkum, L. Feldbrugge, J. P. C. Verhoosel, B. D. van der Waaij, M. van Persie, and H. Noorbergen. Big data analysis for smart farming: results of to2 project in theme food security. *Wageningen University & Research*, 2016.

[86] S. W. Smith. Data compression. https://www.dspguide.com/ch27/5.htm, 2011. [Online; accessed Oct. 2019].

[87] R. Calderbank, S. Jafarpor, and R. Schapier. Compress learning: Universal dimensionality reduction and learning in the measurement domain. *Princeton University*, 2009.

[88] N. K. Boyland. The influence of social networks on welfare and productivity in dairy cattle. https://ore.exeter.ac.uk/repository/bitstream/handle\/10871/19360/BoylandN.pdf?sequence=1&isAllowed=y, 2018.

[89] T. Wey, D. T. Blumstein, W. Shen, and F. Jordan. Social network analysis of animal behaviour:a promising tool for the study of sociality. *Animal Behaviour*, 75:333–344, 2008.

[90] X. Qi, R. D. Duval, K. Christensen, E. Fuller, A. Spahiu, Q. Wu, Y. Wu, W. Tang, and C. Zhang. Terrorist networks, network energy and node removal: A new measure of centrality based on laplacian energy. *J. of Social Networks*, 2:19–31, 2013.

[91] A. Cavanga, I. Giardina, A. Orlandi, G.Parisi, and A. Procaccini. The starflag handbook on collective animal behaviour: Part ii, three-dimensional analysis. *Animal Science*, 76:238–248, 2008.

[92] D. Reynolds. Gaussian mixture models. In *MIT Lincoln Laboratory*, 244 Wood St., Lexington, MA 02140, USA, 2007.

[93] B. McMahan and D. Ramage. Federated learning: Collaborative machine learning without centralized training data. https://ai.googleblog.com/2017/04/federated-learning-collaborative.html, 2017. [Online; accessed Oct. 2019].

[94] G. Y. Wong, F. H. F. Leung, and S. H. Ling. A hybrid evolutionary prepossessing method for imbalance datasets. *Information Sciences*, 454:161–177, 2018.

[95] S. Xu, J. Zhan, B. Man, S. Jiang, W. Yue, S. Gao, C. Guo, H. Liu, Z. Li, J. Wang, and Y. Zhou. Real-time reliable determination of binding kinetics of dna hybridization using a multi-channel graphene biosensor. *Nature Communication*, 2016.

[96] GitHub. Authenticated encryption. https://cryptography.io/en/latest/hazmat/primitives/aead/, 2019. [Online; accessed Oct. 2019].

[97] Individual Contributors. Diffie-hellman key exchange. https://cryptography.io/en/latest/hazmat/primitives/asymmetric/dh/, 2019. [Online; accessed Oct. 2019].

[98] A. Panisson. Generalized random waypoint to support any number of spatial dimension. htps://github.com/panisson/pymobility, 2012.

[99] Inria. Simgrid: Versatile simulation of distributed systems. http://simgrid.gforge.inria.fr/simgrid/latest/doc/, 2017. [accessed Aug. 2019].

Appendix A

Learning in the compressed data domain: Application to milk quality prediction

Journal Title:	Information Sciences
Article Type	Journal
Complete Author List	Dixon Vimalajeewa, Chamil Kulatunga and Donagh P. Berry
Status	Published: vol. 459, pp. 149-167, May 2018

Information Sciences 459 (2018) 149–167

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Learning in the compressed data domain: Application to milk quality prediction

Dixon Vimalajeewa[a,*], Chamil Kulatunga[a], Donagh P. Berry[b]

[a] Telecommunications Software and Systems Group, Arclabs Research and Innovation Centre, Waterford Institute of Technology, Carriganore, Waterford, Ireland
[b] Teagasc, Animal & Grassland Research and Innovation Centre, Moorepark, Fermoy, Co. Cork, Ireland

ARTICLE INFO

Article history:
Received 5 July 2017
Revised 1 May 2018
Accepted 2 May 2018

Keywords:
Compressed learning
MIRS
Principal component analysis
Wavelet transformation
Partial least squares regression
Fog computing

ABSTRACT

Smart dairy farming has become one of the most exciting and challenging area in cloud-based data analytics. Transfer of raw data from all farms to a central cloud is currently not feasible as applications are generating more data while internet connectivity is lacking in rural farms. As a solution, Fog computing has become a key factor to process data near the farm and derive farm insights by exchanging data between on-farm applications and transferring some data to the cloud. In this context, learning in the compressed data domain, where de-compression is not necessary, is highly desirable as it minimizes the energy used for communication/computation, reduces required memory/storage, and improves application latency. Mid-infrared spectroscopy (MIRS) is used globally to predict several milk quality parameters as well as deriving many animal-level phenotypes. Therefore, compressed learning on MIRS data is beneficial both in terms of data processing in the Fog, as well as storing large data sets in the cloud. In this paper, we used principal component analysis and wavelet transform as two techniques for compressed learning to convert MIRS data into a compressed data domain. The study derives near lossless compression parameters for both techniques to transform MIRS data without impacting the prediction accuracy for a selection of milk quality traits.

1. Introduction

Even though smart farming is advancing with the recent developments of Internet of Things (IoT), cloud-based computing, and deep learning, it has become one of the most challenging industrial sectors in big data analytics due to the limitations of ICT infrastructures [47]. However, according to the statistics from the Food and Agriculture Organization of the United Nations (FAO), smart farming will be a key contributor to sustainable intensification in agriculture to feed the 9.2 billion human population by 2050 [1]. There is also a growing interest in pasture-based smart dairy farming in the countries like New Zealand and Ireland, which tend to be in less direct competition with human edible protein and energy sources. Therefore, more harmonized research is needed to optimally utilize ICT infrastructures in precision dairy farming to minimize consumed storage space, communication and computations to facilitate contemporary analytics providing near real-time insights on-farm [37]. This is where the notion of effective data compression approaches are important.

* Corresponding author.
E-mail addresses: dvimalajeewa@tssg.org (D. Vimalajeewa), ckulatunga@tssg.org (C. Kulatunga), Donagh.Berry@teagasc.ie (D.P. Berry).

https://doi.org/10.1016/j.ins.2018.05.002
0020-0255/

Most sensor-based technologies and IoT platforms are designed today to collate and store vast quantities of raw data readings from different sources in geographically distributed farms. Many computational facilities for data analytic applications such as MyAgCentral[1] are now seeking computational resources in cluster-based servers in large centralized data centres. At same the time, the Agricultural Information Management Standards of FAO (AIMS) has already started developing standards and maintaining interoperable trans-national databases for open agricultural data. Therefore farm data will be aggregated as big datasets and there is a requirement to store these data for long-term analytical purposes. This is beneficial since aggregation of data, which extracts a large number of descriptive features in temporally and spatially diverse domains, contributes to an improved learning accuracy. Therefore, compression of such data without a loss of accuracy is vital in terms of the *storage* requirement.

Dissemination of data in its raw format (i.e., in the measurement domain) into large cloud-based data centres is generally not feasible for most farms due to high energy consumption, time criticality of the applications, and the poor/costly rural internet connectivity. For example, if a disease detection system is centralized, it may slow down the farmers' response because of the necessity to transfer vast quantity of data readings to a remote cloud and wait for the outcome to return. Therefore, compression of data is also important in terms of the *communication* efficiency.

However, the key challenge today is whether the centralized storage and computational technologies (with communication networks) contributing to smart dairy farming will still not be sufficient to deliver the future demand without an advanced data analytic infrastructure closer to remote farm management systems. Therefore, a scalable computational infrastructure under constrained resources (proximate to the farm) is essential. In such a constrained infrastructure, compression is a key performance factor also for *computational* efficiency in addition to storage and communication.

Emergence of Fog Computing: With the increase in the amount of data generated from connected sensors, there is a demand to move processing capabilities closer to the data sources, which is in contrast to centralizing raw data in a large data centre. This phenomenon of distributing computations towards the data was first termed as data gravity by *Dave McCrory* in 2010 and is now being realized with new technologies such as Fog (i.e., edge) computing [3] and cloudlets [22]. Fog computing can enable datasets to be processed at the extreme edge of the internet. This computational infrastructure may collectively be formed by low computational proximate devices located near or within the farm. Therefore Fog computing will be a key enabler for many farm analytics to run using scalable in-memory data processing platforms like Spark,[2] Flink,[3] Storm[4] and H20[5] with in-memory databases like Ignite[6] and SAP HANA. Therefore raw-data compression near the data source is a desirable requirement for near future.

As a result, machine learning models, which have targeted highly-provisioned cloud infrastructures, must be re-designed for these resource-constrained infrastructures to minimize storage, communication and computational requirements. New distributed machine learning paradigms like compressed learning [4] and attribute-distributed learning [50] have significant potential to develop effective learning models [49] rather than centralizing all raw datasets from the farms. The main motivation of the present paper is to validate a compressed learning approach [4] for milk quality analysis based on Mid-infrared spectroscopy (MIRS) technology, which can effectively overcome those three challenges in Fog computing. With compressed learning, any machine learning algorithm can be used in a low-dimensional (i.e. latent) space without decompressing data, while optimizing the resource requirement as well as learning efficiency and accuracy of outcomes. Even though the compressed learning approach has been widely used in many fields for learning from complex data sources, such as high-resolution image and video processing and text analysis [26,31], its applicability is new to the MIRS based milk quality analysis.

MIRS is the most economical technology used for assessing milk quality. Therefore, MIRS spectra in predictive models are frequently used to develop farm decision-support tools for efficient milk data processing. For instance, the OptiMIR project has used MIRS of milk recordings in an innovative way to observe different characteristics of cows such as energy balance and early detection of diseases. Also, the routinely obtained MIRS of milk can be used for deriving novel models to quantify milk composition on both an animal basis and on bulk tank samples as well as derive milk related herd-level phenotypes [29]. In addition, variation in MIRS of milk can be used as an indicator in predicting animal characteristics such as the physiological state of an animal and its feed efficiency. The collaborative use of MIRS milk data from different farms can also improve the accuracy of the predictions. Therefore processing vast quantities of milk samples with Fog computing is highly desirable for MIRS milk quality analysis in the future smart dairy farming.

Conventional MIRS analysis [42,43] has been conducted based on co-located data processing by a single computational facility. As shown in Fig. 1, the raw data are directly collated into a repository, mostly by non-experts of data science, and later analysed by the domain-specific data science experts. This significantly increases the computation and power resource requirement on the cloud using raw MIRS data. In modern distributed processing infrastructures, data pre-processing such as de-noising and dimensionality reduction can be carried out closer to data sources. It would, in turn, improve three forms of resource efficiency of the system and reduce the input cost compared to the conventional approach. Water absorbance

[1] https://myagcentral.com/.
[2] http://apache.org.
[3] http://www.h2o.ai.
[4] https://www.keep.eu/keep/project-ext/21114/OptiMIR.
[5] http://succinct.cs.berkeley.edu.
[6] http://CRAN.R-project.org/package=wavelets.

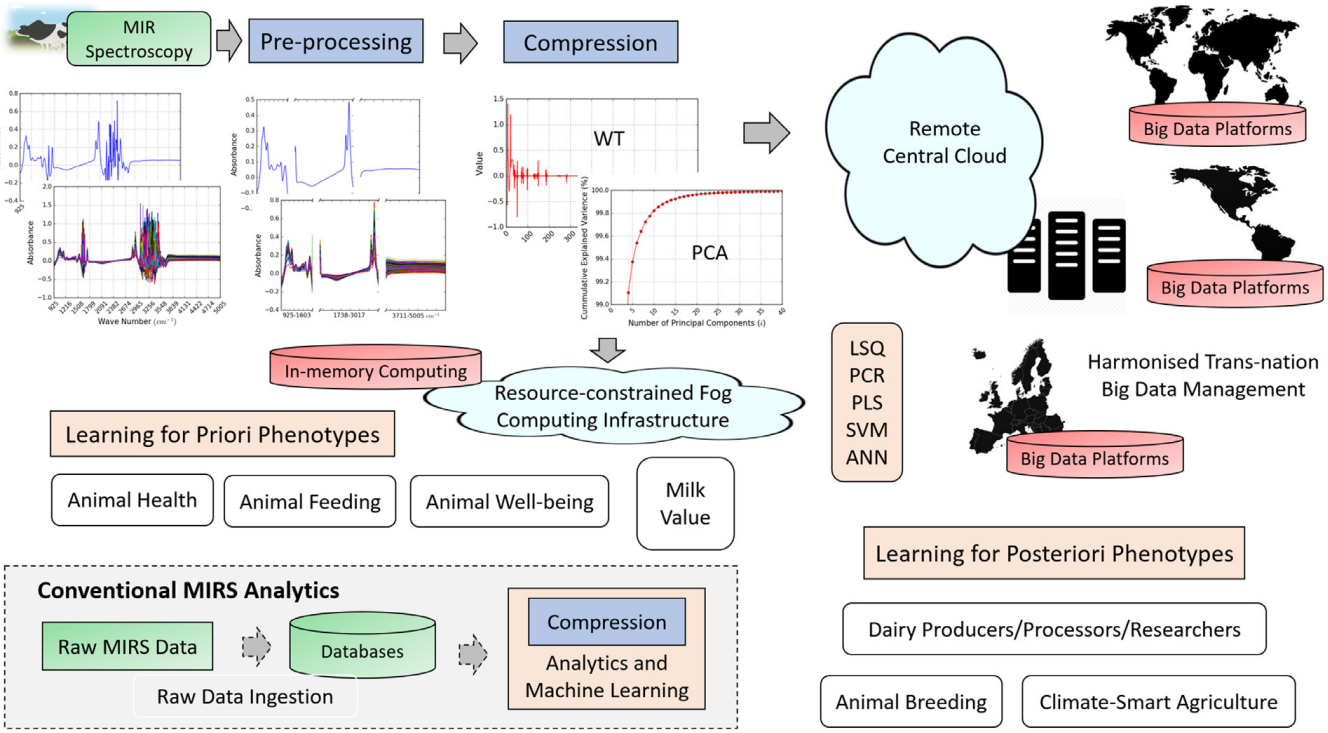


Fig. 1. PCA and WT for MIRS have been applied to avoid overfitting and de-noising in the conventional spectrometry analysis. The two techniques can be used for data compression in future distributed analytics platforms with compressed learning.

data collected using MIRS technology, for instance, hampers the accuracy of milk quality prediction. Removal of these data using distributed computing, prior to sending to the cloud would potentially improve the model accuracy as well as reduce the amount of data in the cloud. Therefore, interpretation of biological data on the edge, using domain-specific knowledge of MIRS, would optimize resource utilization both in big data analytics as well as Fog computing [3].

Compression techniques such as Principal Component Analysis (PCA) [20], Wavelet Transform (WT) [44], manifold and deep learning methods [15] have been widely used for learning from compressed data. The present study investigates the linear learnability of MIRS milk quality data for a selection of milk quality traits in a compressed data domain. We examine in detail PCA and WT as two compression techniques, which have been widely used for MIRS data analysis [42]. The two compression techniques will provide a near lossless compression for many of the currently investigated milk quality parameters. The study concludes with the generalized/harmonized compression parameters required for the two compression techniques to perform compressed learning, i.e. the number of principal components (l) in PCA and the number of wavelet coefficients (r) in WT. We also discuss the additional factors to be required for de-compression, if needed. The impact on the MIRS prediction accuracy at different compression levels was investigated using Partial Least Squares (PLS) linear regression modelling, which has frequently been used in milk MIRS-based predictions [43]. We discuss the importance of sample size in PCA and WT-based compressions and the benefits of supervised compression in compressed learning. Furthermore, we compare our PCA and WT-based compressed learning approaches with state-of-the-art neural network based deep learning techniques such as auto-encoder, GoogleNet and ResNet. While the root mean square error obtained for our approach is comparable for certain features, it is typically higher compared to these techniques. However, the use of deep learning techniques requires a large amount of resources that makes their deployment unsuitable for our resource constrained environments.

Section 1 introduces the paper by discussing the importance of learning in the compressed data domain for MIRS-based analytics from the perspective of Fog computing and big data analytics. Section 2 presents the related works in compressed domain machine learning approaches and applications. Section 3 describes the MIRS techniques used in predicting milk quality traits. Compressibility analysis of MIRS data using PCA and WT is given in Section 4. Section 5 presents the performance statistics of applying the PLS on compressed MIRS data. Section 6 discusses generated results, applicability and a comparison with a state-of-the-art techniques while Section 7 concludes the paper.

2. Related works

The concept of learning in the compressed data domain has been used in a vast range of applications such as hyperspectral image analysis in neuro-science [7] and geo-sciences [41], feature selection in video processing [15,26], machine learning applications in mobile computing [12], distributed data fusions in sensor networks [31,33], as well as classification of complex and big data structures (e.g. text and images) [11,32]. Generally, the primary purpose of using machine learn-

ing in the compressed data domain (in the rest of the paper we refer to simply as compressed learning) was based on a few main reasons: (1) efficient access to large data volumes in big data computations, (2) energy-efficient communications between constrained devices, and (3) computations in resource limited Fog computing environments. In general, the main categories of compressed learning techniques comprise of the PCA, WT, and deep neural networks as compression methods and the learning methods such as regression and classification. The related works presented here have shown that compressed learning has effectively minimized communications, memory, and data storage, while also reducing the learning complexity and hence the processing time of the applications.

A universal framework for compressed learning, in association with compressed sensing, has been presented periodically in the literature [7]. Dimensionality reduction and data compression have been applied on measurement data based on different basis functions mainly using Fourier and Wavelet. To avoid complete reconstruction of time domain signal of electroencephalograms based on random projections, Shoabi et al. [36] provided a comprehensive analysis of a methodology and mathematical framework for compressed learning with data sparsity. Additionally, Lu et al. [27] explained a compressed signal processing approach to adequately preserve the similarity metric of pattern recognition in electroencephalograms. The generality of random projections on Nyquist-domain data enables significant reductions in computation.

In order to accurately reconstruct a signal from the Nyquist-domain, the highest frequency of a signal should be less than half of the sampling rate [38]. However, Donoho in 2004 proposed a compressed sensing approach, stating that with the knowledge of signal's sparsity, a signal can be recovered even with fewer samples [6]. This compressed sensing approach combines signal acquisition and compression into one step (i.e. compression at the time of sampling) instead of performing in two steps (traditional sampling) [38]. Hence, compressed sensing reduces potentially the computational, storage, and communication in higher dimensional data processing compared to the traditional data sampling and compression. Therefore, compressed sensing has gained much attention in the recent past for compressed learning with higher dimensional data such as photography, holography and facial recognition [34].

The requirement of dimensionality reduction of big data for subsequent use in machine learning were discussed and classical PCA has failed as a strategy when the number of observations is very large. This has resulted in issues of memory and storage limitations for single processor computers. As an alternative, Zhang et al. [49] proposed a new PCA approach based on scanning data by rows. The study [8] outlined compressed linear algebra (CLA) for in-memory operations such as matrix-vector multiplication in compressed data domain. Also Elgohary [8] documented the drawbacks of heavy weight compression algorithms due to computational complexity in decompressing and lightweight algorithms because of poor compression ratios while making a clear case for the operation of linear algebra directly using matrices with compressed data.

Learning from feature extraction has been extensively used in image and video analysis. A novel technique for constructing high resolution images from low resolution images and recognition of such images using a singular value decomposition (SVD) based PCA approach have been investigated in [15]. Moreover, a SVD-based approach to extract potential global features from facial images given in [16] used special properties of singular values of an image to devise a compact, global feature for image-representation. Also the authors of Jian et al. [16] theoretically proved that leading singular values can be used as rotation-shift-scale-invariant global features of an image. Texture image retrieval and classification based on SVD was investigated in [18], while Guo et al. [11] proposed a texture-based image classification approach based on cross-covariance matrix of image textures. The authors of Guo et al. [11] claim to have reduced the processing time of image classifications by using the compressed domain cross-covariance vectors of the original image data. Sometimes PCA and WT have been jointly used in compressed learning. For instance, PCA has been used to accelerate WT and eye location verification based on the features extracted from facial images using WT-salient maps in [17]. Moreover, a WT-based salient feature extraction approach has been presented in [14]. Another approach to mimic the human visual system's salient detection in images using wavelet-based salient patch detection is given in [19].

High resolution space-borne optical images were analysed when proposing an efficient ship detection approach using compressed learning with a Deep Neural Network (DNN) algorithm [41]. Only the relevant information was extracted using WT from space-borne optical images to observe ship positions with less detection time. Similarly, Perera and Mo [33] proposed an energy-efficient ship navigation method based on compressed domain learning. A large amount of sensor-based ship navigation data was compressed on-board using PCA. Also Perera and Mo [33] applied regression analysis in an on-shore located data centre to derive optimal navigation paths based on the compressed data. A lossless dictionary-based compressed learning approach for unsupervised feature learning for text data was discussed in [32] as well as also the applicability of k-grams based compressed data for many tasks in text processing.

Compressed learning has been used extensively in feature learning applications in sequential video frames. The method presented in [26] can separate (as background and foreground of a sequence of video frames) a large set of raw data using a small amount of information based on prior knowledge. The authors of Loung [26] named the protocol Compressive Online Robust Principal Component Analysis (CORPCA) and stated that it can be used to extract only significant features from high dimensional data of time-variant processes by taking a single instance at a time (i.e., a frame). In CORPCA, compression is performed recursively using the compressive information that is extracted from its previous stage. In [31], PCA was applied in compressed domain for re-enforcement learning. This approach reduced the table sizes of state space and action space thereby minimizing the memory spaces and learning times. The study [31] also showed that PCA reduced communications in multi-agent distributed learning environment.

Compressed learning and models are becoming more popular in big data and mobile computing platforms. DNN models are commonly used in mobile applications. However, such applications are too large to fit into constrained mobile computing resources. Therefore, compressed versions of DNN models were introduced in [12], which have the same properties as their corresponding original models and provide an energy-efficient platform to run those models. In addition, distributed computing frameworks such as Apache Spark have been combined with a compressed data representation framework developed by the *Saccinct* project. This framework enables to query data stores in a compressed data domain, so that Spark users benefit in searching point queries directly on a compressed representation of the input data. Deep learning techniques have achieved higher classification accuracy than the traditional compression techniques. An application of auto-encoder technique for hyper-spherical image classification has been provided in [48].

Different spectrometric analyses have benefited from compression techniques to perform further learning. The study [28] discussed a general overview of MIRS applications as a phenotyping tool for deriving milk quality traits, while Gunrdeniz and Ozen [10] explained the use of MIRS with WT and PCA in a quantification of the extra-virgin olive oil adulteration process. MIR (and NIR) spectroscopic techniques were used in [24] to determine the dry matter content of tea; WT and PLS chemometric techniques were performed to determine the tea dry matter content.

In general, compressed learning has been used extensively in a vast range of spectrometry applications. However, spectrometry analysis towards a generalized (harmonized) compressed learning process for MIRS-based milk quality monitoring has not been thoroughly investigated. Even though the two compression algorithms, PCA and WT, have been used [42], application of data compression where data are generated (near sensing) has not been thoroughly studied for use in future analytics platforms.

3. MIRS for predicting milk quality traits

Fourier Transform (FT)-MIRS is the prominent MIRS approach currently used in routine milk testing. Globally, milk samples from individual dairy cows and bulk samples are routinely taken to assess milk quality which can be subsequently used by dairy producers and processors in making management decisions but also by breeders to identify genetically elite candidate parents of the next generation. The milk quality information originates from predictions from the transmittance of light in the mid-infrared region (i.e. 2500–25000 nm; 900–5000 cm^{-1}) of the electromagnetic spectrum. The outcome of the MIRS analysis is a spectrum for each sample and this transmittance value is available for each wavelength irrespective of its information content [29]. Milk protein, fat, lactose, urea, minerals, acetone, ketone bodies, casein are some of the most reported milk quality parameters predicted from MIRS [29], which are used in deriving many priori and posteriori phenotypes by the stakeholders.

Some compression algorithms like Lempel–Ziv–Welch (LZW) deliver the objective of data compression (i.e., data are compressed without losing information), but de-compression is still necessary to convert the data into its original (measurement) domain because statistical learning cannot be applied to compressed data. This type of compression method help in optimizing issues such as storage and communication difficulties. However, de-compression brings back the original data dimensionality with irrelevant and redundant MIRS data. Thus, the complexity of learning from the original data remains unchanged. Therefore, such lossless compression algorithms increase only the computational cost of de-compression without making any contribution to the learning process. In conventional cloud systems, decompression happens in high-end servers where energy and computational power are generally not a constraint. However, in Fog computing, decompression may be performed at a resource constrained Fog node [3]. Therefore, compared to the general compression-decompression approaches, the compressed learning concept in MIRS using PCA and WT offers an effective methodology in a resource constrained infrastructure.

The quality and the dimensionality of MIRS data are crucial factors for machine learning. High dimensionality and multicollinearity (i.e. correlated data) also limits the use of multiple linear regression. As an early approach to compressed learning, Cands and Wakin [5] proposed a better lossless data compression technique using only a few and potentially disjoint sets of highly significant WT coefficients in an orthonormal basis. In addition to using WT, in the present study, we attempted to compare compression performances with PCA as a widely used technique in MIRS. There are many well-defined pre-treatment techniques (e.g. scaling, scatter correction, etc.) in MIRS analytics to undertake quality control of data which we need to apply before the compression process. For example, the spectrum contains dissolved water absorbance ($O = H$ bonds) in the 1500–1800 cm^{-1} and 2900–3800 cm^{-1} ranges, and these regions are not useful in the prediction of milk quality traits.

In compressed learning, the original data can be recovered with the recovery algorithm of PCA and WT only if it is needed. However, by performing all analytical processes in the compressed domain, we can eliminate the additional cost of de-compression, which is in contrast with some other user-interacted data compression applications like in multimedia. Since WT uses a known basis (*Daubechie* – 4 at scale q in Fig. 2), no additional information is needed to decompress. However, in PCA, the loading matrix and the column averages (a and P in Fig. 2) of the original data matrix are required for the decompression as we explain later in the paper.

The compression level of a MIR spectrum (X) depends on the target response variable (γ) of the learning algorithm. For example, an analytical engine for animal health status can be run in one computational sub-system while another analytical algorithm for milk quality may be run in another sub-system. In the present study we investigate a generalization approach of compression of MIRS data only using X (Fig. 2), which is the important research question in compressed learning for MIRS

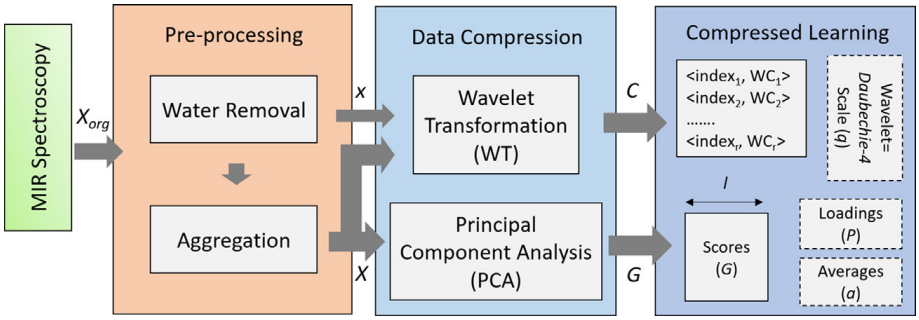


Fig. 2. Pipeline of the Compressed Learning framework: Data pre-processing/aggregation is performed at a very early stage. Compression of MIRS data is carried out using PCA or WT irrespective of the intended milk quality trait (unsupervised). Data in the compressed domain (i.e. scores in PCA or indexed WCs in WT) will be used by different machine learning applications.

milk quality monitoring. However, we briefly discuss the possibility of further compression based on a known γ within the discussion section.

The data used in the paper originated from the Teagasc research dairy farm at Moorepark, Ireland where MIR spectra were collected and the composition of milk was determined using FOSS MilkScan prediction equations. The input data matrix contained the spectra of 712 different milk samples in the wavenumber region 925–5005 cm^{-1} with a resolution of 3.853 cm^{-1}; wavenumbers were rounded to the nearest integer. As a result, the given spectrum contained 1060 transmittance data points. Therefore, the original MIRS spectra used (called gold standard) to develop linear prediction models was a (712×1060) size matrix and denoted by X_{org}. We converted them to absorbance values by taking log_{10} of the reciprocal of the given transmittance values. Absorbance indicates the amount of absorption of electromagnetic radiation when the MIR light penetrates through the milk sample. Higher absorbance values indicate that the MIR light penetrates less at certain wavenumbers according to the molecular bonds. In addition, percentages of the selected milk nutrient components; lactose, fat, protein and urea, corresponding to each sample were stored in a column matrix ($\gamma_{n \times k}$, where $n = 712$ and $k = 5$). PLS model calibration and validation were applied on to these gold standard data (γ) to derive our generalized compression parameters.

4. Compressibility of MIRS milk quality data

This section will discuss the compressibility (unsupervised or general) of MIRS dataset using PCA and WT. First, we will investigate the data redundancy of the available MIR spectra and hence the compressibility of such data, which can be improved using PCA and WT without noticeable information loss. Second, we will discuss the selection of main input (or compression) parameters required for the PCA compression (number of principal components (l)) and WT compression methods (number of wavelet coefficients (r)). Since there is no prior information regarding the learning purposes (e.g. regression, classification) which the compressed data will be used for, the compression should be performed by preserving the original properties of the MIRS data as much as possible. Therefore, the selection of compression parameters is important and should be performed carefully. In order to select reliable values for l and r, their impact on the quality of compression was examined. The variance explained by the principal components and the reconstruction error were used to quantify the quality of compression. We have used compression ratio as the final evaluation metric of our approach as it indicates computation, communication and storage performance of the analytics infrastructure. The notations used to represent different matrices, vectors, and values in MIRS dataset, PCA, WT and PLS algorithms are given in Table 1.

4.1. Pre-processing of the MIRS data

In spectrometry analysis, dissolved water adds unnecessary variability to the MIR spectra and could affect the resulting prediction accuracy. Most possibly, this effect is a random fluctuation or a systematic shift of the spectra. For instance, milk spectrum indicates two random sharp fluctuation regions, which occur in the wavenumber regions 1500–1800 cm^{-1} and 2900–3800 cm^{-1} per visual observation. Those regions are the water absorbance regions according to the pure water spectrum at 25°C. In distributed analytics, we precisely identify those two regions based on PLS model calibration on our gold standard data and suggest these regions should be removed in the pre-processing stage before the compression. Therefore, based on our systematic identification of the two water regions, the corresponding wavenumbers can be removed from all raw MIR spectra in the measurement data domain.

In order to identify the water regions, we selected visually observable bare minimum water regions as 1464–1849 cm^{-1} and 2890–3814 cm^{-1} and removed these from X_{org}. Then we progressively recaptured one wavenumber at a time from the discarded regions to our predictors. In each step, the impact of the addition was quantified based on cross-validated root mean squared error ($RMSE_{CV}$) of the PLS predictive algorithm (explained in Section 5). The predictive error indicated a noticeable increase as the water absorbance regions began to be included in the prediction. Our finalised wave regions removed were 1607–1734 cm^{-1} and 3021–3707 cm^{-1}. By removing the wavenumbers which were in the water absorbance

Table 1
Mathematical notations used in the paper to represent MIRS dataset and PCA, WT, PLS algorithms.

Notation	Description
X_{org}	Original MIR spectra with values in absorbance
X	Water removed MIR spectra
n	Number of samples in the gold standard
m	Number of wavenumbers in X after removing water
γ	Target variables of milk quality components in %
k	Number of selected milk quality components
(x, y)	A sample (a row) of X and γ in the gold standard
l	Number of PCs selected using PCA
G	Scores matrix of PCA in compressed domain
P	PCA loadings matrix for data recovery
a	Averages of selected l columns
r	Number of WCs selected using WT compression
C	WT after thresholding in compressed domain
q	Level of scale in WT
X'	Reconstructed MIR spectra from the compressed domain
u	Number of Latent Variables in PLS

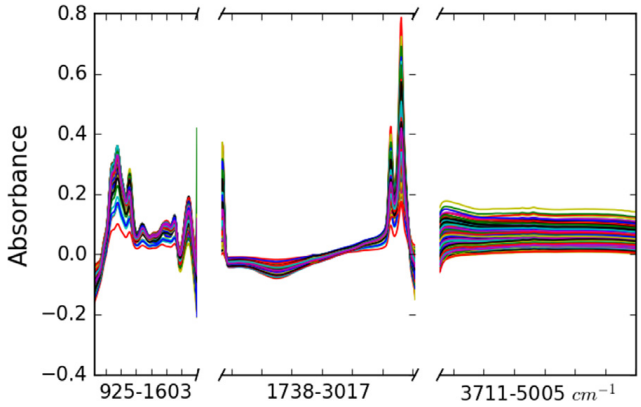


Fig. 3. Water-free MIR spectra (X) of 712 milk samples in the wave region 925–5005 cm^{-1}. Pre-processing has reduced the feature-space dimensionality from 1060 wavenumbers to 847 due to removal of water absorption.

regions, the dimensionality of water free spectrum (X) became 712×847 (i.e., $m = 847$), which reduced the amount of unwanted MIRS data by 20.1% and Fig. 3 represents the water absorbance regions removed spectra. The pre-processing stage could precisely remove the wavenumbers from the original spectra to obtain X, which is then fed into the compression stage.

In order to reliably develop our prediction model, we applied pre-treatment processes to the gold standard data. We mean-centred and scaled the values of X so that the mean and the standard deviation (SD) of each wavelength was 0 and 1, respectively. Scaling was not a compulsory approach in MIRS data since all the features were in the units of absorptions. However, this standardization could avoid confusion when using widely available machine learning libraries. We verified the normality of each response variable using *Shapiro* similarity check as a pre-requirement for applying PLS regression [35]. Outliers in γ, which were identified as when the difference between the value and its mean is more than three times the SD of a target variable, were removed from the data. Gold standard data (i.e., γ) were not available for all the samples and were therefore not considered if missing. After applying these pre-treatments, the final number of samples used for fat prediction was 701 and for lactose, protein and urea was 704.

4.2. Compression with principal component analysis

The existence of strong correlations among the feature vectors makes MIRS predictions unamenable to simple analytic techniques like multiple linear regression due to matrix singularity (n=712 < m=847) and multi-collinearity (correlations among feature variables), which could contribute to over-fitting. To overcome these issues, mostly PCA has been used for dimensionality reduction in MIRS, while WT has been used for de-noising [39]. However, both techniques can also be used for de-noising as well as for dimensionality reduction. In addition, PCA can particularly be useful as a data visualization tool and for feature extraction while WT can be used as an accelerating tool for efficient feature extraction by PCA. Therefore, the order of applying the techniques in a resource-constrained distributed computational infrastructure is important, but this has not been a concern for users of MIR spectroscopic analytics in the past.

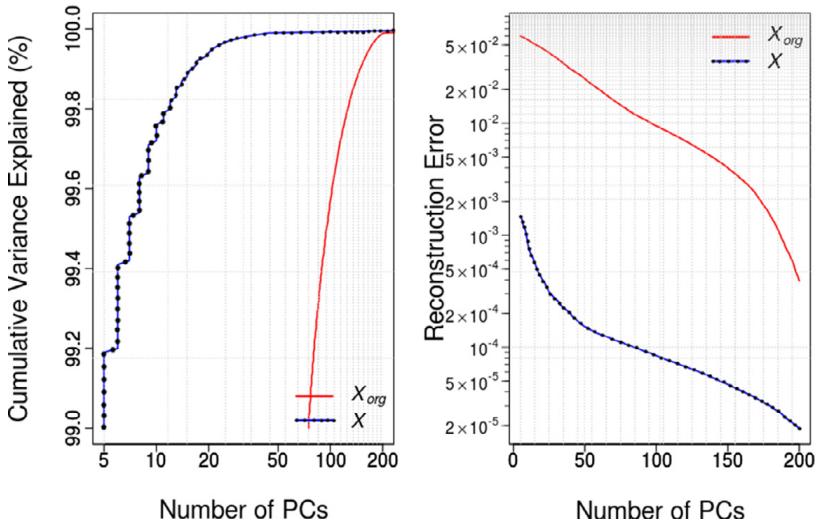


Fig. 4. Cumulative variance explained and the reconstruction error at different number of PCs of the original (X_{org}) and water removed (X) spectra.

Application of PCA in most of the higher dimensional data studies was variance based [45]. Feature vectors, which explain a significant portion of the variance in the original data (motive to capture only significant information as possible), are extracted based on the correlations among the different predictive variables (columns of X). Once a certain number of PCs are selected, this forms a low dimensional subspace of data such that every selected component is orthogonal to the other with minimum loss of information. Because of neglecting components, which contribute little to explaining the variability in the data, this concept has been used for dimensionality reduction of multi-dimensional data in a vast range of applications [20].

From a mathematical point of view, suppose $n < m$ in the feature matrix $X_{n\times m}$, where n and m are integers. PCA computes a new set of transformed variables called principal components (PCs) as linear combinations of the original variables. The first PC ($PC1$) accounts for the largest possible variance in X while the second PC ($PC2$), which explains the second largest variance in X, is computed to be orthogonal to $PC1$. The third PC ($PC3$) is derived to be orthogonal to both $PC1$ and $PC2$. The remaining PCs are computed in the same way and the transformed values of these PCs are called scores. The total number of PCs that can be generated from X is the minimum of n and m. In our MIRS data, $PC1$, $PC2$ and $PC3$ respectively explained 6.9%, 5.6% and 4.8% of the variability in X_{org}, and 68.5%, 23.0% and 4.9% of the variability in X. The Singular Value Decomposition (SVD) technique was used in the present study to compute PCs [20].

The new feature space of X ($G_{n\times l}$ - compressed domain data) is formed by selecting the columns from G which correspond to the first l largest singular values in D. The value of l is decided upon based on a threshold of the cumulative variance of PCs. The coefficients of the linear combinations are contained in $P_{l\times l}$, which we need to transform $G_{n\times l}$ back to the original domain X' (i.e. when the column average vector a of length l is provided).

We used the R package *pls 2.6-0* [30] which was developed to calculate the PCs from our MIRS dataset. PCA of X_{org} and X gives 712 PCs, which is the minimum of $n = 712$ for both 1060 or $m = 847$. The proportion of variance accounted by the different number of PCs were studied and also the loss of information from recovering the original data from those PCs were quantified by using the reconstruction error for both X_{org} and X. Fig. 4 (left) shows the cumulative percentage of variance from 99% onwards explained by PCs of both X_{org} and X. Fig. 4 (right) also shows the reconstruction error $(\sum_{i=1}^{n}(\sqrt{\sum_{j=1}^{m}(\delta x_{i,j}^2)}/m)/n)$ at different PCs, where $\delta x_{i,j}$ is the difference between the original value (X) and the reconstructed value (X') of a data point for $i = 1, 2, \cdots, n$ and $j = 1, 2, \cdots, m$. For example, the number of PCs needed to explain 99.9% of total variance were 145 and 20 for the original and water removed spectra, respectively. Then the dimensions of the compressed domain data of X can be reduced to 712×15 providing a compression ratio (defined as $\frac{c}{m} \times 100$, where $c = l, r$, for the rest of the paper) of 98.2% having a reconstruction error of 3.05×10^{-4}. The cumulative variance explained by the PCs of X was above 99.99% and its increment was less than 10^{-3} after the first 100 PCs. Therefore, the optimal value for l was selected as 100 with minimum loss of information (0.01%) in our analysis. The reconstruction error corresponding to the first 100 PCs of X was 6.28×10^{-5} which guaranteed that the amount of information loss was small (the compression ratio was 85.96%).

Fig. 4 shows that PCA can significantly reduce the dimensionality of the MIRS data at different accuracy levels. The results also show that the water-related wavelengths contribute a significant amount of variability in the dataset, which should be removed based on our concluded wavenumber regions prior to compression. Hence, a significant amount of communication and computation energy can be saved for the benefit of future Fog and big data analytics. PCA-compressed data also minimizes over-fitting where the compressed domain data ($G_{n\times l}$) can directly be used in subsequent linear regression models.

However, our presented PCA compressed data may not have removed high frequency noise, while Wavelet compression in the next section can remove such noise in MIRS data. Since PCA is an unsupervised learning approach, it only accounted for collinearity among feature variables. However, in most of the real-world datasets, including our MIRS data, collinearity between response and feature variables also exist. In such situations, supervised dimension reduction techniques can be used and the optimal number of PCs required to generate a stable prediction model might be further reduced as shown in Section 6.

4.3. Compression with wavelet transformation

WT can be applied to a single or any finite group of spectra and analysed on any scale with orthogonal basis functions [44]. Every basis function consists of two types of functions : (1) wavelet function (mother wavelet), which is a high-pass filter capturing sharp behaviours (called details), and (2) scaling function (father wavelet), which is a collection of scaling functions capturing more general behaviours (called approximations) and act as a low-pass filter. In general, the data passes through these two filters and then generates approximate and detailed signals at a certain scale (q). The outcome of the high-pass filter is taken as Wavelet Coefficients (WCs), representing high frequency components. When the scale is higher, WCs are increased while the Scaling Coefficients (SC) are reduced. The number of filtering steps might deteriorate the transformed signal and may affect reconstruction (de-compression) after a certain scale [24], which we need to select for our MIRS data compression.

When selecting a basis function, the important properties to be considered are orthogonality of basis, preservation of data sparsity, independence between wavelet coefficients, and easiness in the reconstruction of the signal. Since there are different types of basis functions such as Haar, Symmetric and Daubechie, selecting an optimal basis is an important factor in WT. For instance, Haar wavelet is not suitable for the description of smooth functions; instead we used Daubechie-4 in our evaluations with the most commonly used WT, which is Discrete Wavelet Transform (DWT) [42].

Let x be a signal (e.g. a spectra) from X of length m. First we apply zero padding (which may sometime cause a considerable edge effect which linear padding minimizes [2]) to extend the array of $n = 847$ to 1024, which is the nearest 2^{10} format of our dataset (to apply WT, the signal length must be of the form 2^b, where $b \in \mathbb{Z}^+$). DWT was applied on a Daubechies-4 wavelet basis for different number of scales where the maximum was 10. The elements which are less than a selected threshold (λ) were regarded as noise (insignificant information) and removed from the transformed signal. According to Artime [2], there are many thresholding methods such as universal, hard, and soft, but we used the soft thresholding approach. We then obtained our compressed domain data matrix $C_{n\times r}$. The indexes of the selected components are required to reconstruct the original data. In reconstructing the original signal x, we replaced all the removed positions with zero and applied the Inverse DWT for the same numbers of q. We used Multi Resolution Analysis (MRA) [43], which is a simple, fast and easily illustratable DWT method.

In general, MIR spectra contain high and low frequency signal components. The signals that have frequencies above a certain level are considered as noise components. We used the R package *wavelets 0.3 in our MRA based DWT*. Wavelet transform was applied on X and the coefficients were retained, which has the dimensions of 712×1024. The number of SCs and WCs are shown at the 4th scale in Fig. 5 for a single spectrum. We used a threshold (λ) of 0.01 to compress the spectrum at this level discarding insignificant components. According to this threshold, scaling and wavelet coefficients of 53 and 74, respectively can be selected. These components need to be stored as key-value pairs at the compression stage to use in compressed learning.

The optimal number of scale levels (q) and threshold (λ) are the main parameters required for selecting the most significant WCs in WT. Therefore, the behaviour of the number of significant WCs were experimented with by changing the values of q and λ. The first graph of Fig. 6 shows the variability in the number of WCs at different scales under different threshold values (exponentially selected between 0.0012 and 0.02). For simplicity, we refer to the number of WCs as the sum of both scaling and wavelet coefficients at a certain threshold in the rest of the paper. According to Fig. 6, the number of coefficients is high (but with lower reconstruction error) for small thresholds. Increasing q up to the maximum possible scale is not required. Fig. 6 shows a saturation behaviour at the number of WCs after the 6th scale, which we will use in Section 5. Therefore, WT does not capture any high or low frequencies of spectra after this point in our MIRS data. For example, at a threshold of 0.01, our data can be compressed to 127 coefficients with a 1.9×10^{-3} reconstruction error. To select an optimal value for r, the reconstruction error was computed for different WCs using the scale and threshold values of 6 and 0.0025. The second graph in Fig. 6 represents the behaviour of reconstruction error. The reconstruction error of X was almost saturated (the error change was less than 10^{-3}) after 200 WCs. Therefore, the optimal r value was selected as 200 with the reconstruction error of 8.2×10^{-4} and a compression ratio of 71.91%.

Our MIR spectra can be considerably compressed while keeping most of the critical information and discarding most of the unnecessary information both using PCA or WT techniques. This concludes therefore that spectra can be transformed into their compressed domain and can be recovered with minimal error, if necessary. However, our results show that PCA required a fewer number of components than the required number of coefficients in WT to achieve a similar reconstruction error. The next section will investigate the impact of our compression on the PLS prediction accuracy of four different milk traits and hence derive our generalized/harmonized compression parameters (l and r) for compressed learning.

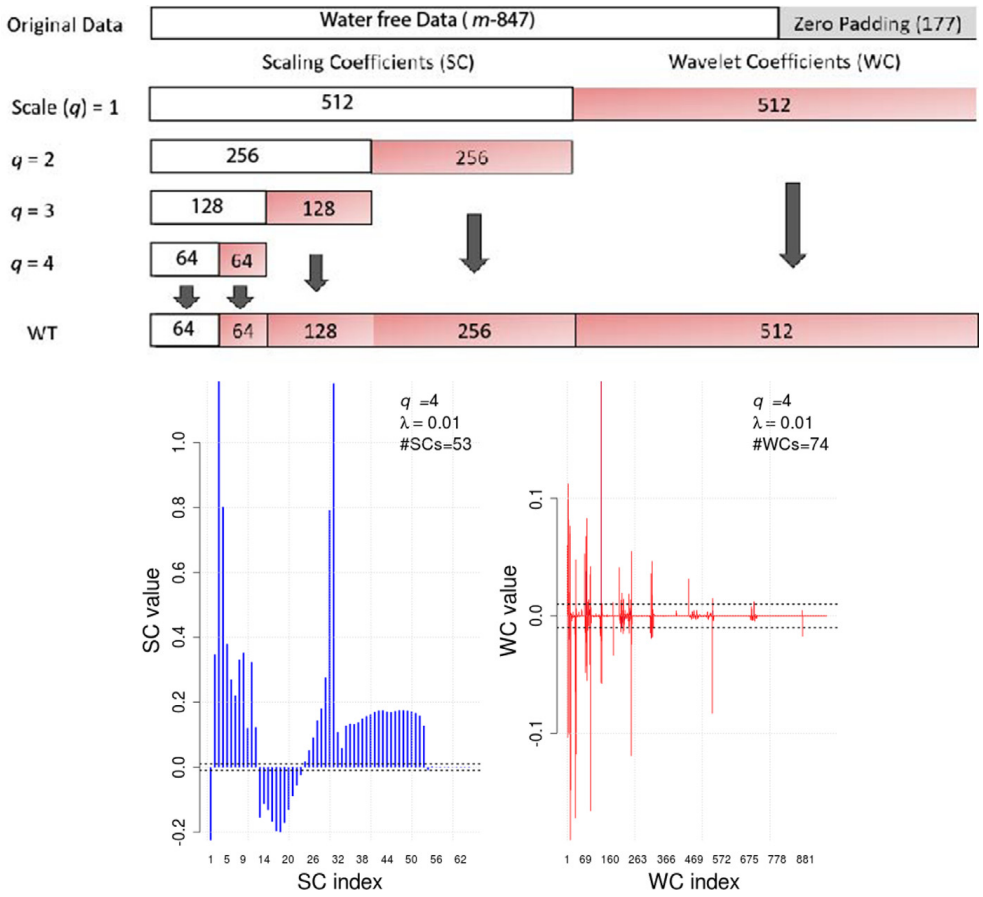


Fig. 5. Distribution of SCs and WCs at the 4th scaling ($q = 4$) of a single water-free spectrum of our MIRS data using 'Daubechies-4'. Threshold (λ) of 0.01 indicate that the spectrum compresses to 127 components.

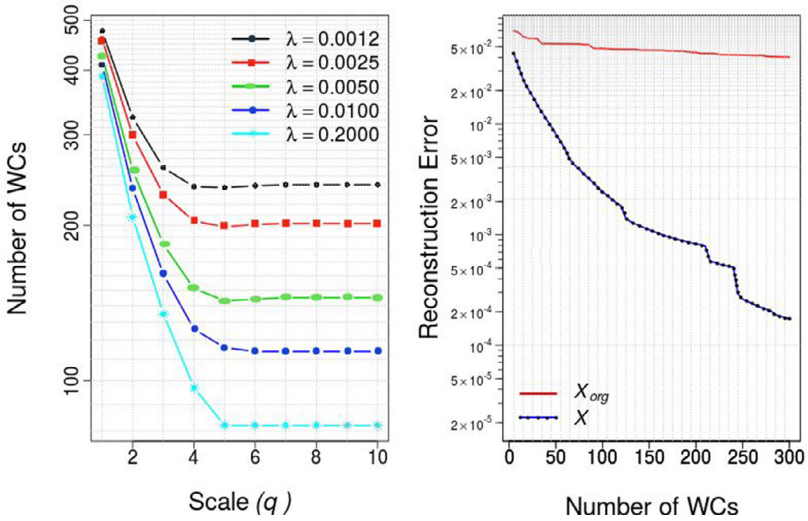
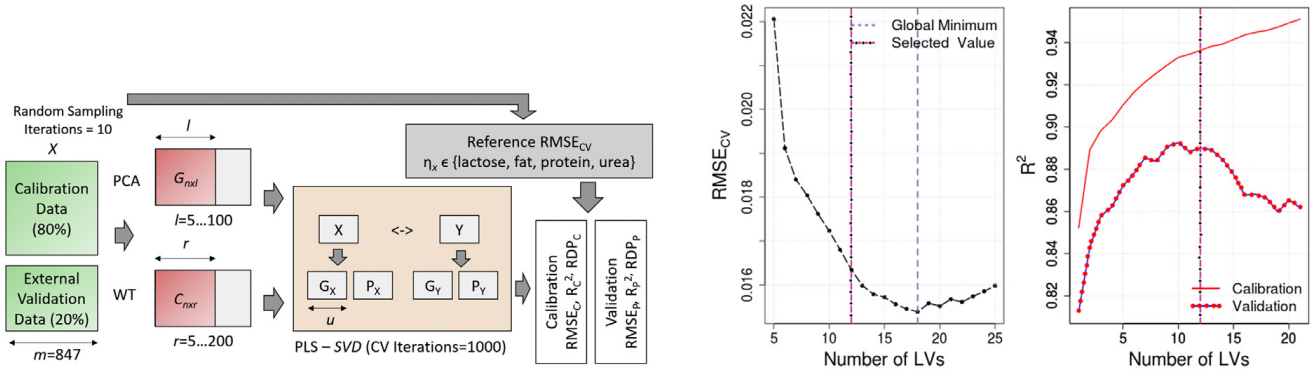


Fig. 6. Number of significant WCs at different thresholds (λ) and different scales (q). The number of coefficients saturate at scale 6. Reconstruction errors in WT are higher when compared to PCA.

5. Impact on prediction accuracy by compressed learning

This section investigates the impact of compression parameters on compressed learning performances. First, we study the impact of l and r on the learning performances derived from a supervised compressed learning approach and second, we select optimal parameter values based on their impact on the learning performances. We apply PLS, which is commonly used for analysing MIRS data [1,10,28], on the compressed MIRS data (i.e. PCA scores $G_{n\times l}$ and Wavelet-transformed data $C_{n\times r}$) to quantify how much the predictive accuracy is impacted by PCA and WT based compressions. At different compression



(a) Overview of the different stages when PCA and WT compressed data is applied with the PLS regression (Green and pink colours respectively represent the data in the measurement and the compressed domains)

(b) Determination of the optimal number of LVs at an optimal $RMSE_{CV}$ for lactose using uncompressed data. The optimal value is selected not to exceed $RMSE_{CV}$ of 0.001 from the absolute minimum

Fig. 7. Overview of the PLS learning procedure from PCA and WT and selection of LVs from PLS calibration for building predictive models. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

levels (i.e., varying l and r), prediction performance in model calibration and external validation using compressed data is compared with the data in the uncompressed measurement domain (X). The following indexes for the regression model have been used to evaluate the compressed learning performances.

The root mean-square error ($RMSE$) quantifies the standard deviation of the residuals (between the real and the predicted response variable γ) and is shown in the units of absorbance. The coefficient of determination (R^2) depicts the proportion of variance in the response variable γ explained by the predictor variables in X. These measures are computed by the following two equations (subscript i - real response value and p - predicted value).

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(y_i - y_p)^2}{N-1}}, \quad R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - y_p)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2}$$

The Ratio Performance Deviation (RPD) represents the practical utility of the model, and is calculated as $(1 - R^2)^{-1/2}$. As a rule of thumb, if $RPD > 3$, then the model can be used for practical analytical purposes.

All performance indexes are calculated for both the calibration (c) and external validation (prediction) (p) data segments of our gold standard MIRS data. Based on these evaluations, near lossless compression parameters l and r for PCA and WT, respectively are derived for each milk trait. We have selected four of the most used milk quality parameters: lactose, fat, protein and urea, all derived from milk MIRS [28].

5.1. Partial Least Squares (PLS) regression

PLS is a projection method that models the relationship between the predictors X and responses γ (a.k.a. Projection on Latent Structures) [9]. The PLS method considers not only the correlations among the predictor variables in X, but also the correlations each predictor in X and the response in γ. The general procedure of PLS is somewhat similar to when dimensionality reduction of PCA is combined with Least Squares Regression (LSQ), which is called as PCR. However, PLS and PCR differs mainly in the methods used in extracting factor scores. PCR produces a loading matrix P reflecting the covariance structure among the predictor variables. PLS produces a loading matrix P reflecting the covariance structure between the predictor and the response variables [9]. The set of significant components in PLS is called the Latent Variables (LV). PLS decomposes both X and γ using SVD.

Fig. 7(a) shows the logical overview of the essential steps that we have followed in this section to derive l and r. The sub-sampled training dataset (model calibration) is selected randomly having 80% of the total n samples. The remaining set of samples is used for testing the model (external validation). To increase the validity of model performance, we repeat the above process for 10 different data selections while keeping the same ratio for training and test data partitions. We have selected samples randomly (from $n = 712$) under each iteration and the average of performance measures has been calculated.

5.2. PLS accuracy using uncompressed MIRS data

First, we calculated PLS accuracy with the MIRS data in the measurement domain (X). This accuracy ($RMSE_{CV} = \eta$) was used as the reference to estimate our near lossless compression parameters. Different compression parameters; l_x and r_x where $x \in \{lactose, fat, protein, urea\}$, were derived by fitting a PLS model on the compressed data for the four selected milk parameters. We achieved the prediction performance using compressed data to be comparative with the reference

Table 2
PLS model performance on the original (X). Our near lossless PCA and WT compressions find optimum number for l and r according to these reference values.

Milk trait	Reference values		Calibration			External validation		
	#LVs (u)	$RMSE_{CV}$ (η)	$RMSE_c$	R_c^2	RPD_c	$RMSE_p$	R_p^2	RPD_p
Lactose (X_{org})	5		0.0173	90.25	3.2152	0.0190	88.53	3.0379
PLS model performance for water removed Spectra (X)								
Lactose (X)	12	0.0154	0.0151	92.60	3.6929	0.0167	91.17	3.4578
Fat	5	0.0892	0.0865	88.49	2.9540	0.0919	87.35	2.8607
Protein	4	0.0601	0.0574	76.12	2.0570	0.0625	73.61	2.0461
Urea	15	0.3443	0.3098	81.55	2.3350	0.3523	77.64	2.1428

model performance (η_x). The selection procedure of the compression parameters is explained only for lactose but the same procedure was followed for fat, protein and urea and the summary is given.

First, a PLS regression model was fitted to the training data in the uncompressed domain (X) and we obtained the cross-validated mean of $RMSE_{CV}$ by changing the number of LVs in the PLS model to select the minimum error at an optimal number of LVs (u). However, in this process, u was selected as the LV corresponding to the $RMSE_{CV}$, which did not make a considerable difference (p-value ≤ 0.001) to the global minimum of $RMSE_{CV}$ (i.e., LV corresponds to the selected $RMSE_{CV}$ such that the difference between the selected and the global minimum $RMSE_{CV}$ is not greater than the p-value). u has been selected (as explained above) in our evaluations according to a permutation model explained in [30] and a 10-fold cross validation, followed 1000 iterations, for selecting each LV.

According to Fig. 7(b), the optimal $RMSE_{CV}$ of 0.0154, is achieved with 12 LVs for lactose ($u_{lactose} = 12$), when the water-removed spectra (X) were used. Twelve LVs were selected as the optimal number of LVs even though the absolute minimum of $RMSE_{CV}$ occurred at 17 LVs. The graph on the right of Fig. 7(b) provides performance statistics of the PLS model with a comparison of calibration and external validation statistics based on R^2. In Fig 7(b), the performance indexes of the calibration and external validation values do not change much beyond the selected optimal LV point (after the dashed line). Therefore, the optimal PLS model can reliably be derived with 12 LVs for lactose.

Table 2 presents the optimal $RMSE_{CV}$ and minimum LVs we can achieve with the measurement domain data for all the four different milk traits we have selected. Statistics in Table 2 indicate a well performed regression models for lactose and fat, because the R^2 were $>87\%$ for both models and the RPD was ≥ 3 for lactose and close to three for fat in both the calibration and validation. The regression models of protein and urea content were not as good as the lactose and fat regression models, because R^2 and RPD values were only $>73\%$ and ≥ 2 in the both the calibration and external validation.

Table 2 also shows that we can achieve a 12.7% improvement in $RMSE_{CV}$ concurrent with a 20.1% compression by just removing the water-related wavelengths of the spectrum during the pre-processing stage of our compressed learning. Then we conducted the same PLS procedure using PCA and WT compressed data by changing compressed dimension parameters l and r .

5.3. Impact on PLS accuracy with PCA compression

The performances of the PLS model were computed by changing the number of PCs, $l = 5, \cdots, 100$ with a step of 5 PCs. The results on predicting lactose are given in Fig. 8(a). During the evaluations, the score matrix ($G_{n \times l}$) at different selected number of PCs was applied as the compressed domain input to the PLS model. $RMSE_{CV}$ of cross-validation were compared with the reference accuracy of lactose ($\eta_{lactose}$), which is given in the Table 2.

PLS calibration and validation accuracies using PCA compressed data decreased as the number of PCs increased. With 45 PCs, it shows a similar minimum $RMSE_{CV}$ compared to the reference PLS accuracy of 0.0154. Adding more PCs after 45 PCs into to the model did not make a significant contribution to improve the model performance (i.e. the impact of l on lactose predictive model is up to a certain value only). Thus, the results reveal that the PCA compression with at least 45 PCs is stable. Therefore, we conclude that the optimal compression level can be achieved with 45 PCs for lactose prediction ($l_{lactose} = 45$) resulting in a compression ratio of 94.7%.

Results in Table 3 shows the optimal number of PCs required to predict all the milk traits using PLS. These models were derived in the similar way to as described for lactose. Moreover, different milk traits have their own optimum number of PCs; $l_{lactose} = 45, l_{fat} = 30, l_{protein} = 37$ and $l_{urea} = 65$. Therefore, with respect to each trait, the water-removed spectra can be compressed by 94.7%, 96.5%, 95.6% and 92.3% for lactose, fat, protein and urea, respectively using PCA at the compression stage.

5.4. PLS accuracy with WT compressed data

The same procedure of PLS regression as explained in the previous section for PCA compression was applied for the WT compressed data. In this case, the PLS was applied to the WT coefficient matrix $C_{n \times r}$ by changing the number of WCs, $r = 5, \cdots, 200$ with a step of 5 WCs. Fig. 8(b) shows PLS prediction performance for lactose and the regression model with

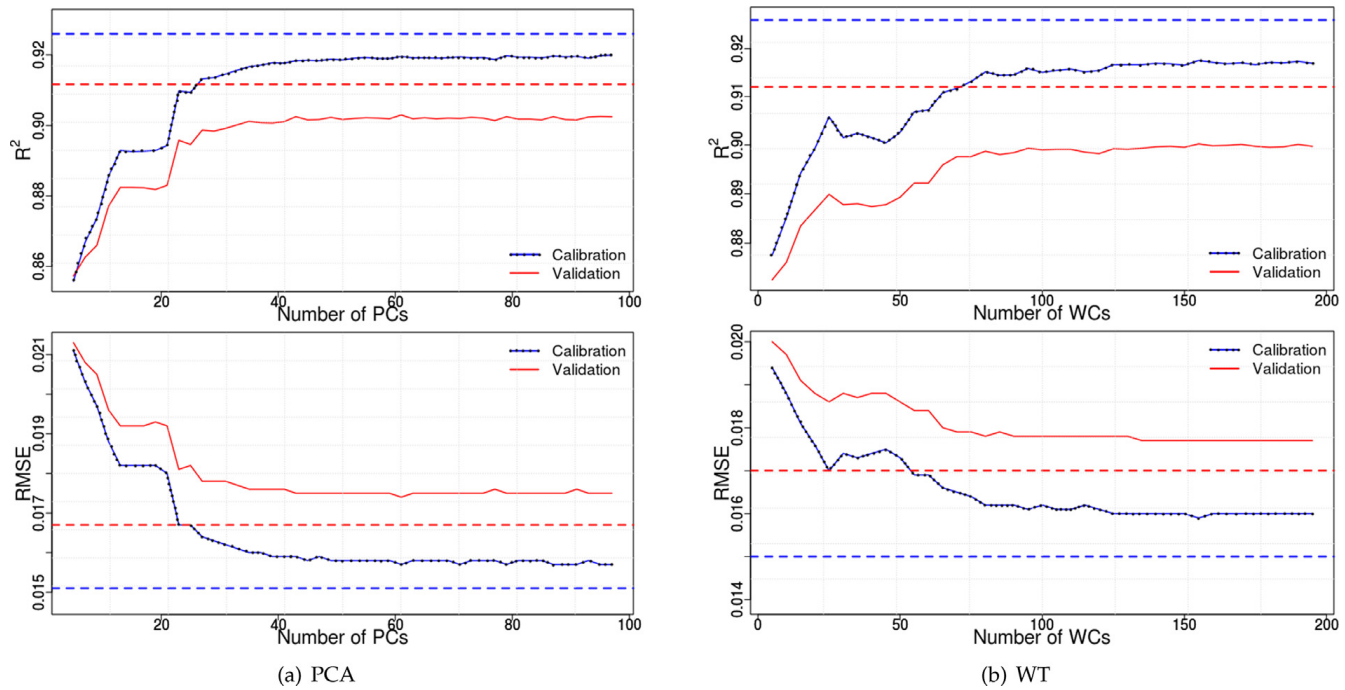


Fig. 8. Compressed domain PLS performance at different number of PCs and WTs for lactose. The dashed line represents the optimal PCA (a) and WT (b).

Table 3
PLS model accuracies for the selected milk traits at optimal PCA compressed points. Optimal number of PCs has been selected based on 0.01 $RMSE_{CV}$ threshold from the absolute minimum. Optimal $RMSE_{CV}$ has been tallied to reference η.

Milk trait	#PCs (l)	#LVs (u)	Calibration			External validation			Reconstruction error
			$RMSE_c$	R_c^2	RPD_c	$RMSE_p$	R_p^2	RPD_p	
Lactose	45	12	0.1580	91.85	3.5152	0.0175	90.16	3.3018	1.266×10^{-4}
Fat	30	5	0.0871	88.30	2.9386	0.0918	87.41	2.8537	1.860×10^{-4}
Protein	37	4	0.0577	75.91	2.0474	0.0627	73.50	2.0448	1.554×10^{-4}
Urea	65	15	0.3334	78.63	2.1687	0.3705	75.07	2.0351	0.928×10^{-4}

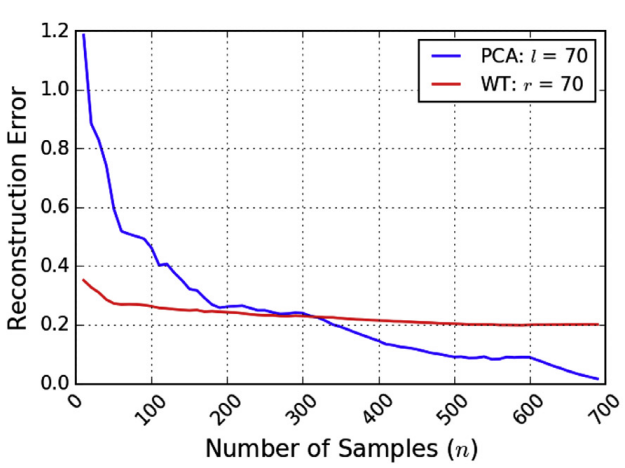
Table 4
PLS model performance for different milk traits for Wavelet compressed data. Optimal number of PCs has been selected based on 0.01 $RESE_{CV}$ threshold from the absolute minimum. Optimal $RMSE_{CV}$ has been tallied to reference η.

Milk trait	#WCs (r)	#LVs (u)	Calibration			External validation			Reconstruction error
			$RMSE_c$	R_c^2	RPD_c	$RMSE_p$	R_p^2	RPD_p	
Lactose	70	12	0.1650	91.31	3.3803	0.0178	89.86	3.2284	4.8×10^{-3}
Fat	40	5	0.0864	88.49	2.9682	0.0920	87.37	2.8455	4.3×10^{-3}
Protein	45	5	0.0575	76.08	2.0549	0.0625	73.72	2.0432	9.5×10^{-3}
Urea	75	15	0.3333	78.64	2.1700	0.3730	74.95	2.0239	3.9×10^{-3}

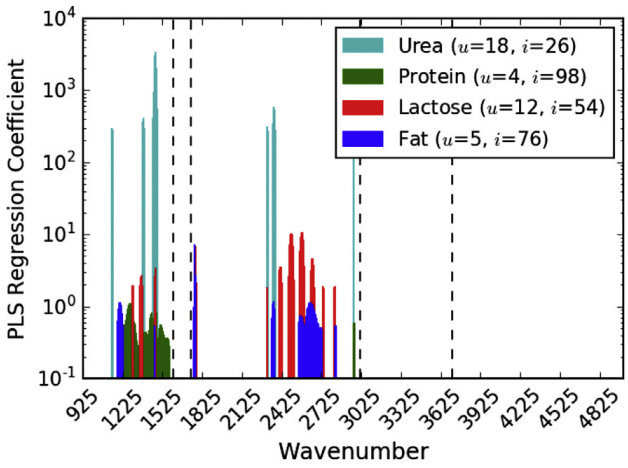
100 WCs indicates an $RMSE_{CV}$ close to the data domain accuracy of $\eta_{lactose}$. Therefore, the optimal compression was achieved using 100 WCs for the prediction of lactose ($r_{lactose} = 70$). In addition, the behaviour of the impact of r was also similar to the behaviour which was obtained with PCs in Fig. 8(a).

Table 4 shows the prediction performance in the WT compressed domain for all the selected milk traits. Different milk traits had their own optimum number of WCs; $r_{lactose} = 70, r_{fat} = 40, r_{protein} = 45$ and $r_{urea} = 75$. WT can compress MIRS data by 91.7%, 95.3%, 94.7% and 91.1% for lactose, fat, protein and urea, respectively.

PLS regression models focused on finding an optimum level of compression (optimal l and r) for our MIRS data based on either PCA or WT. We validated the near lossless compression using its impact on PLS regression-based learning accuracies for the different milk traits. Therefore, transformed data can be used to learn in their compressed domain. Both PCA and WT compressions had similar compression performance. Based on the four milk quality traits we selected, the number of PCs in a general PCA compression (l) and the number of WCs in a general WT compression (r) should have at least $l = 65$ and $r = 75$ components (i.e., 92.3% and 91.1% compression can be achieved from PCA and WT, respectively). Therefore, selection of the largest number of PCs and WCs is the requirement to preserve the predictability of urea without losing any information on the investigated milk traits.



(a) Batch-based compression of PCA and WT selecting different number of samples ($n = 10 \cdots 700$)



(b) Significant PLS coefficients (higher than the stand deviation) at the optimal number of LVs.

Fig. 9. Compressed Learning with sample-size sensitivity and supervised compression.

6. Discussion

6.1. Sample size selection of PCA and WT

Real-time data transfer always consumes greater energy and is not used in many agricultural infrastructures. Instead delay-tolerant networks and data logging systems are mostly used [21]. Therefore, the MIRS source can collect a certain number of spectra before data compression and transmission takes place (e. g. in robotic milking cows are milked in every 7–10 minutes by a single machine). If the delay is large, some extra memory space is needed to store the spectra until data are compressed and later transmitted. However, there can also be cases where in-situ milk quality (online) monitoring is used by the dairy industry. In this case, time becomes a critical factor and WT should be used for compression instead of PCA.

The sample size (n) plays an important role in PCA-based compression since fewer samples create instability in PCA. The general understanding is that the larger the sample size, the better the stability. Selecting an adequate sample size for our MIRS data is a compromise for timeliness of decision making. There are no simple rules to determine the appropriate sample size for PCA. The variability in reconstruction error with respect to sample size was examined with PCA and WT compressions for our MIRS data X. According to Fig. 9(a), WT using our recommended number of WCs, does not improve reconstruction error as the number of spectra available increase. PCA using the recommended number of components can improve reconstruction error by increasing the number of samples. At a certain point beyond 190 samples, PCA has less reconstruction error than WT.

6.2. Customization using supervised compression

Standard PCA does not know what portion of variance in each variable is important and should be preserved. Sufficient application knowledge with intended milk traits (supervisory learning) can further optimize our compression performance. PLS can be used in supervised compression only using LVs (compressed domain) or using significant wave indexes in a linear model (measurement domain). We analysed the composition of each milk quality trait within the spectrum using PLS (Fig. 9(b)). The results show that a customized approach can be applied at the compression stage or on top of our generally compressed data using PCA or WT to further improve our compression performance.

As an example, a farm decision support tool may need to identify only the fat and protein content of certain milk samples [29] to quantify cow-level energy balance in the herd. Such a customized system can further compress MIRS data beyond our unsupervised compression techniques, when data are transferred between the Fog nodes or into the big data systems.

6.3. Impact on advanced analytics

Linear PCA assumes that the original data can be converted into a single scale and the relationship between the orthogonal PCs are linear. However, these assumptions are not always true with real data. For instance, categorical data consists of ordinal and nominal variables, which is not easy to convert onto a single scale. Hence, PCA compression could possibly lose significant information, due to multi-scale data with non-linear behaviour and correlations. If data have non-linear behaviours, linear PCA may inadequately capture significant variances. According to Linting [25], non-linear PCA overcomes not only these issues, but also facilitates the application of PCA without changing the existing scales. Even though some

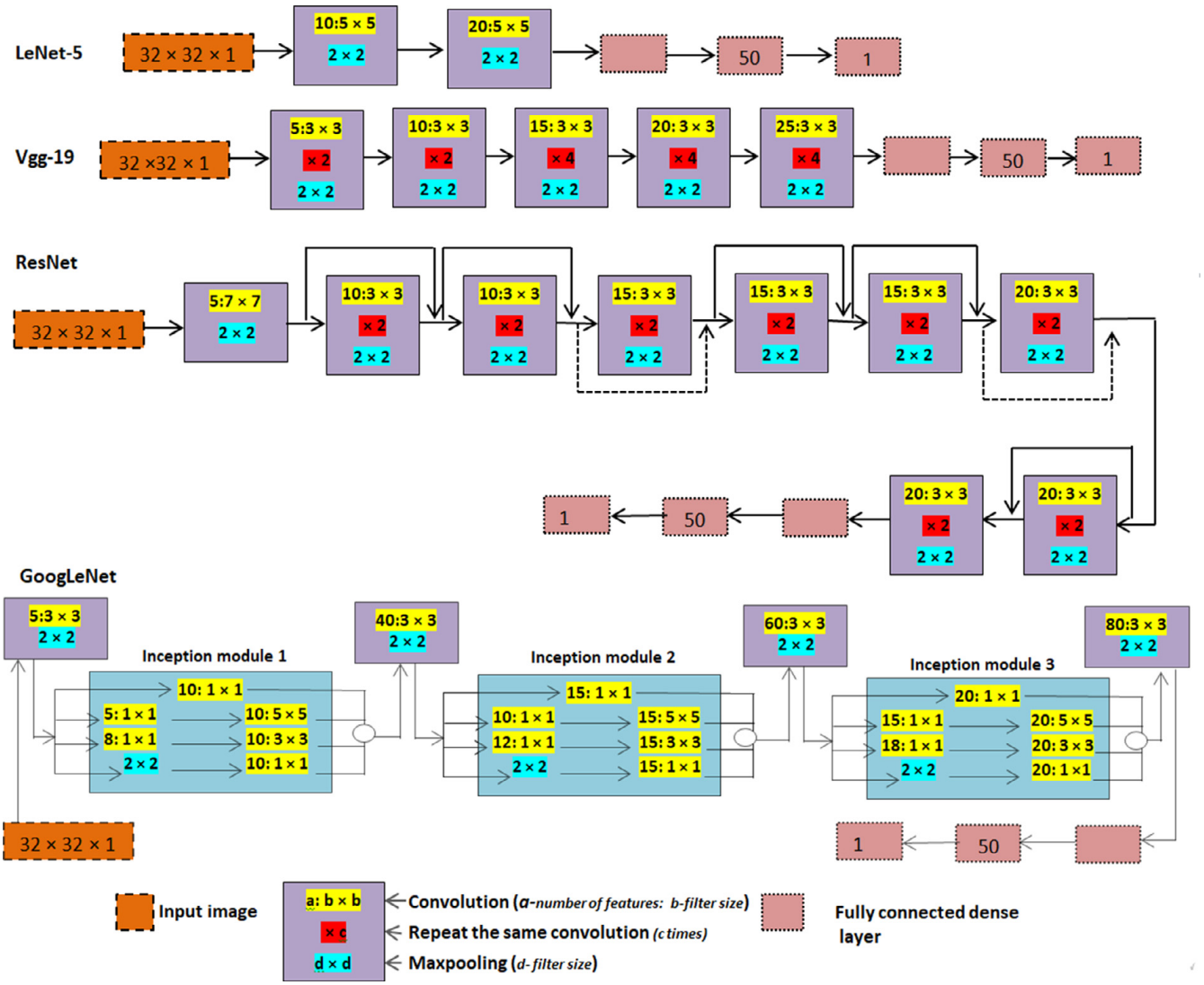


Fig. 10. Network architectures of four state-of-the art deep learning techniques LeNet-5, Vgg-19, GoogLeNet, and ResNet. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

PCs capture very little variance from the data, those PCs may represent substantial information. Therefore, PCA variants such as kernel PCA may solve some of these difficulties in linear PCA, where compressed learning with MIRS needs further investigation.

PCR and PLS predictive methods are commonly used for statistical learning processes in spectrometric analytics. However, these methods fit a linear regression model. If compressed domain data presents a non-linear behaviour, those linear models would not contribute to derive best fit stable predictive models. Use of linear models may create a negative impact on the robustness and accuracy of the learning process. Therefore, advanced methods such as Support Vector Machine (SVM) [39] and Artificial Neural Network (ANN) [23] are available (with the improvement of pervasive computational capabilities) and can be used to address non-linear behaviours in MIRS data subjected to the fact that we have preserved non-linearity in the compressed domain data.

6.4. A comparison with state-of-the-art techniques

We have compared PCA and WT compressed learning performances with deep auto-encoder (DEA) [48], LeNet-5, Vgg-19 , GoogLeNet, and ResNet [13,40], using our MIRS data, all of which are emerging deep learning techniques. The LeNet-5, Vgg-19, ResNet, and GoogLeNet can be considered as the extensions of the LeNet model. These techniques are un-supervised and different forms of convolution neural network (CNN) models, which can also be considered as lossy compression techniques yet differ from the engineered compression techniques (e. g. JPEG, LZW). The PCA and WT are faster, simpler, and require less computational power, but considered only linear properties in the data. Whereas the deep learning techniques are much flexible and able to achieve more precise outcomes than PCA and WT based learning approaches by accounting for the non-linearity in the data. However, for instance, higher model complexity and computational requirements are the

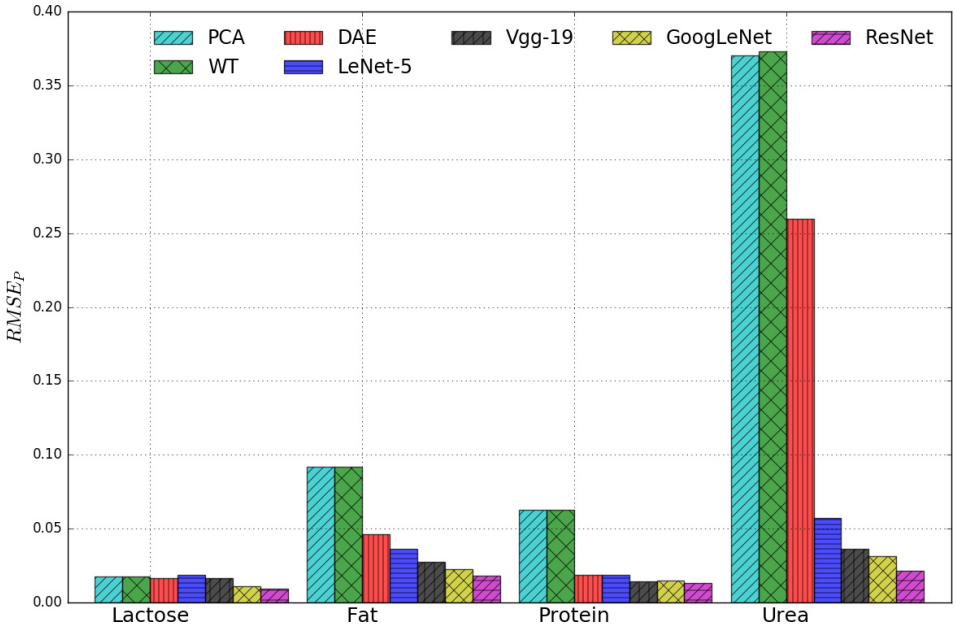


Fig. 11. Comparison of compressed learning performances of PCA and WT with four state-of-the-art deep learning methods LeNet-5, Vgg-19, GoogLeNet, and ResNet.

main implementation constraints in the deep learning approach. To overcome these issues, more advanced versions of CNN approaches have been proposed and the techniques mentioned above are a few of them.

The water-removed MIRS data was used for deep learning. Three encoding layers were used in the deep auto-encoder (DAE) model. The number of decoding layers was same as the number of encoding layers. PLS-based learning procedure was followed to quantify compressed learning performances as in Section 5. The compressed dimension was set to 70 as a middle compressed dimension to the highest feature variables (65-Table 3 and 75-Table 4), which were observed from PCA and WT based compression, respectively. Fig. 10 shows the LeNet-5, Vgg-19, ResNet, and GoogLeNet network models, and to apply these models to our data, each sample was re-sized as a $32 \times 32 \times 1$ matrix, applying zero padding. The convolution layers mostly have 1×1 and 3×3 filters and 2×2 Maxpooling filters. The convolution and pooling operations were performed in the intermediate layer (purple colour box) and pooling was applied after the convolution. The red colour box was removed from the intermediate layer when the same convolution was not repeated. We did not use a dropout layer before making the fully connected layers. Each model has two fully connected layers (second fully connected layer has 70 neurons) and the last dense layer is a regression layer. The network architectures given in [13] were followed to configure the Vgg−19 and ResNet models. Although the same convolution was repeated for six times in the ResNet model in [13], we did it only for four times. The solid and dashed lines in the ResNet model represent the shortcut connection with same and increased dimensions, respectively. When the dimension was increased with stride 2, zero padding and 1×1 convolution were used to match dimensions. Three inception modules (the inception module with dimension reduction [40]) were stacked together to form the GoogLeNet model (for more details about these network configurations, please follow [13,40]). Each model was trained for up to 1000 iterations using the ADAM optimizer and mean squared error loss function. Also, we used a fixed learning rate of 0.01 and the rectified activation function. Finally, to compare the performance of these deep learning techniques with our compressed learning outcomes, the $RMSE_P$ was computed, applying all these techniques to each milk quality parameter.

The Fig. 11 shows the predictive learning accuracies from each deep learning model, including PCA and WT. The learning performances from all methods were approximately similar for lactose and an improvement was observed for fat, protein, and urea. This can be due to the existence of non-linear associations in the MIRS data, which has an impact on predicting fat, protein, and urea in milk [46]. The predictive accuracy also increased with the increasing depth of network models so that the ResNet model achieved the greatest accuracy. Due to the small data size, selecting a sufficient number of features in convolution, and over-fitting were the major issues when training these models. Therefore, learning performances may even improve further by using larger datasets with a comprehensive study of different factors such as data pre-processing, proper constraints, optimizers, and network design.

These state-of-the-art techniques can also be used for the compressed learning which we have discussed in this study and performed well compared to the traditional methods. However, employing them under some circumstances such as with low complexity and under limited computational resources may not be feasible for applications such as distributed data processing using Fog computing, which is one of our main concern in the smart farming industry. These limitations would be minimized by using the ResNet and GoogLeNet. The GoogLeNet model has the potential to control the computational cost required with deep networks so that, it can be used even with limited resources and low-memory requirements [40].

The ResNet model is easy to optimize and gain accuracy by increasing the depth and width of the network [13]. Although more reliable outcomes can be derived efficiently from these deep learning methods, further studies are essential to study the feasibility in employing these techniques in the smart dairy industry because the resources, such as computational infrastructure, energy, and lack of data might still be the major constraints to run these advanced algorithms. As we can see in Fig. 11, the learning performances from all methods are approximately similar for lactose, it may not necessary to apply deep learning for deriving a predictive model for lactose. Thus, performing an initial study to get an overall idea about the general characteristics such as non-linear associations in the original data would help to select the most suitable compressed learning approach. Consequently, we can optimize the utilization of available resources and obtain reliable outcomes in resource constraint environment such as Fog Computing.

7. Conclusions

In this paper, we have shown that MIRS data can be pre-processed and compressed effectively near the data source without impacting the prediction accuracy of most measured milk quality traits. PCA can generally be compressed to 65 principal components and WT can be compressed to 75 wavelet coefficients, which leads to compression ratios of 92.3% and 91.1%, respectively. At these compression levels, PLS using PCA and WT compressed data (i.e. 65 significant scores in PCA and 75 significant coefficients in WT) can achieve the same accuracy, as PLS can achieve using the pre-processed data in the original measurement domain. Therefore, the results show that the compressed learning with MIRS is highly advantageous both in Fog and big data processing, which can preserve communication and computation energy, minimize required memory and storage spaces, reduce application latency and preserve scarce rural network bandwidths.

Acknowledgement

This research was supported by Teagasc (grant no. 13/1A/1977) and Science Foundation Ireland (SFI) (grant no. 13/1A/1977) through the project "PrecisionDairy-*Using Precision Technologies, Technology Platforms and Computational Biology to increase the Economic and Environmental Sustainability of Pasture-based Production Systems* (ID: 13/1A/1977)".

References

[1] N. Alexandratos, J. Bruinsma, World agriculture towards 2030/2050, in: Food and Agriculture Organization in the United Nations, EAS, 2012, pp. 12–21. working Paper No 12-03
[2] C.E.C. Artime, On-line estimation of fresh milk composition by means of vsi-nir spectrometry and partial least squares method (pls), in: IEEE Instrumentation and Measurement Technology Conference, 2016, pp. 1471–1475, doi:10.1109/IMTC.2008.4547275.
[3] R. Buyya, C. Mahapatra, V. Leung, M. Chen, P. Sahu, Fog computing: internet of things realize its potential, IEEE Comput. Mag. 49 (8) (2016) 12–116, doi:10.1109/MC.2016.245.
[4] R. Calrebank, S. Jafarpor, R. Schapier, Compressed learning: universal dimensionality reduction and learning in the measurement domain, 2009.
[5] E.J. Candşs, M.B. Wakin, Introduction to compressive sampling, IEEE Signal Process. Mag. 25 (2) (2008) 21–30, doi:10.1109/MSP.2007.914731.
[6] D.L. Donoho, Compressed sensing, IEEE Inf. Theory 52 (4) (2006) 1289–1306, doi:10.1109/TIT.2006.871582.
[7] M.F. Duarte, Y.C. Eldar, Structured compress sensing: from theory to application, IEEE Trans. Signal Process. 59 (9) (2011) 4053–4085, doi:10.1109/TSP.2011.2161982.
[8] A. Elgohary, Compressed linear algebra for large-scale machine learning, VLDB Endowment 9 (12) (2016) 960–971, doi:10.14778/2994509.2994515.
[9] P.H. Garthwaite, An interpretation of partial least squares, Am. Stat. Assoc. 89 (425) (1994) 122–127, doi:10.2307/2291207.
[10] G. Gunrdeniz, B. Ozen, Detection of adulteration of extra-virgin olive oil by chemometric analysis of mid-infrared spectral data, Food Chem. 116 (2) (2009) 519–525, doi:10.1016/j.foodchem.2009.02.068.
[11] J. Guo, B. Song, F. Jian, H. Qin, Texture classification with cross-covariance matrices in compressive measurement domain, Signal Image Video Process. 10 (8) (2015) 1377–1384, doi:10.1007/s11760-016-0902-9.
[12] S. Han, H. Mao, W.J. Dally, Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding, in: International Conference on Learning Representations, 2016. arXiv: 1510.00149v5.
[13] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778, doi:10.1109/CVPR.2016.90.
[14] M. Jian, K.-M. Lam, J. Dong, Image retrieval using wavelet-based salient regions, Imaging Sci. J. 59 (4) (2011), doi:10.1179/136821910X12867873897355.
[15] M. Jian, K.-M. Lam, J. Dong, A novel face-hallucination scheme based on singular value decomposition, Pattern Recognit. 46 (11) (2013) 3091–3102, doi:10.1016/j.patcog.2013.03.020.
[16] M. Jian, K.-M. Lam, J. Dong, Face-image retrieval based on singular values and potential-field representation, Signal Process. 100 (2014) 9–15, doi:10.1016/j.sigpro.2014.01.004.
[17] M. Jian, K.-M. Lam, J. Dong, Facial-feature detection and localization based on a hierarchical scheme, Inf. Sci. 262 (2014) 1–14, doi:10.1016/j.ins.2013.12.001.
[18] M. Jian, K.-M. Lam, J. Dong, Illumination-insensitive texture discrimination based on illumination compensation and enhancement, Inf. Sci. 269 (2014) 60–72, doi:10.1016/j.ins.2014.01.019.
[19] M. Jian, K.-M. Lam, J. Dong, Visual-patch-attention-aware saliency detection, IEEE Trans. Cybern. 45 (8) (2015), doi:10.1109/TCYB.2014.2356200.
[20] V. Klema, The singular value decomposition: its computation and some applications, IEEE Trans. Autom. Control 25 (2) (1980) 164–176, doi:10.1109/TAC.1980.1102314.
[21] C. Kulatunga, Opportunistic wireless networking for smart dairy farming, IEEE IT Prof. Mag. 19 (2) (2017) 16–23, doi:10.1109/MITP.2017.28.
[22] C. Kulatunga, K. Bhargava, D. Vimalajeewa, S. Ivanov, Cooperative in-network computation in energy harvesting device clouds, Elsevier J. Sustainable Comput. (2017).
[23] L.J. Lancashire, C. Lemetre, G.R. Ball, An introduction to artificial neural networks in bioinformatics - application to complex microarray and mass spectrometry datasets in cancer studies, Briefing Bioinf. 10 (3) (2008) 315–329, doi:10.1093/bib/bbp012.
[24] X. Li, L. Luo, Y. He, N. Xu., Determination of dry matter content of tea by near and middle infrared spectroscopy coupled with wavelet-based data mining algorithms, Comput. Electron. Agric. 98 (2013) 46–53, doi:10.1016/j.compag.2013.07.014.
[25] M. Linting, Nonlinear principal components analysis: introduction and application, J. Psychol. Methods 12 (3) (2007) 336–358, doi:10.1037/1082-989X.12.3.336.

[26] H.V. Loung, Incorporating prior information in compressive online robust principal component analysis, in: arXiv:1701.06852, 2017.
[27] J. Lu, N. Verma, N.K. Jha, Compressed signal processing on nyquist-sampled signals, IEEE Trans. Comput. 65 (11) (2016) 3293–3303, doi:10.1109/TC.2016.2532861.
[28] M.D. Marchi, V. Toffanin, M. Cassandro, Invited review: mid-infrared spectroscopy as phenotyping for milk quality traits, Dairy Sci. 97 (3) (2014) 1171–1186, doi:10.3168/jds.2013-6799.
[29] S. McParland, D.P. Berry, The potential of fourier transform infrared spectroscopy of milk samples to predict energy intake and efficiency in dairy cows, Dairy Sci. 99 (5) (2016) 40564070, doi:10.3168/jds.2015-10051.
[30] B. Mevik, R. Wehrens, Introduction to the pls package, R Project, 2016. http://cran.r-project.org
[31] A. Notsu, Information compression effect based on pca for reinforcement learning agents' communication, International Symposium on Advanced Intelligent Systems, 2012, doi:10.1109/SCIS-ISIS.2012.6504999.
[32] H.S. Paskov, R. West, J.C. Mitchell, T.J. Hastie, Compressive feature learning, Advances in Neural Information Processing Systems 26 (NIPS 2013), 2013.
[33] L.P. Perera, S. Mo, Machine intelligence for energy efficient ships: abig data solution, in: International Conference on Maritime Technology and Engineering, 2016, pp. 143–150, doi:10.1201/b21890-21.
[34] S. Qaisar, R.M. Bilal, W. Iqbal, M. Naureen, S. Lee, Compressive sensing: from theory to applications, a survey, Commun. Netw. 15 (5) (2013) 443–456, doi:10.1109/JCN.2013.000083.
[35] N.M. Razali, Y.B. Yah, Power comparison of shapiro-wilk, kolmogorow-smirnov, lilliefors and anderson-darling test, Stat. Modell. Anal. 2 (1) (2011) 23–33.
[36] M. Shoabi, N.K. Jha, N. Verma, Signal processing with direct computations on compressively sensed data, IEEE Trans. Very Early Scale Integr.(VLSI) Syst. 23 (1) (2015) 30–43, doi:10.1109/TVLSI.2014.2301733.
[37] J. Steenveld, H. Hogeveen, Characterization of dutch dairy farms using sensor systems for cow management, Dairy Sci. 98 (1) (2015) 709–717, doi:10.3168/jds.2014-8595.
[38] T. Strohmer, Measure what should be measured : progress and challenges in compressive sensing, IEEE Singal Process. Lett. 19 (12) (2012), doi:10.1109/LSP.2012.2224518.
[39] A. Subasi, M.I. Gurosy, Eeg signal classification using pca, ica, lda and support vector machine, Expert Syst. Appl. 37 (12) (2010) 8659–8666, doi:10.1016/j.eswa.2010.06.065.
[40] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, D.A. S. Reed, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, doi:10.1109/CVPR.2015.7298594.
[41] J. Tang, C. Deng, G. Hung, B. Zhao, Compressed-domain ship detection an spaceborne optical image using deep neural network and extreme learning machine, IEEE Trans. Geosci. Remote Sens. 53 (3) (2015) 1174–1185, doi:10.1109/TGRS.2014.2335751.
[42] J. Trygg, N. Kettaneh-Wold, L. Wallba, 2d wavelet analysis and compression of on-line industrial process data, Chemometrics 15 (4) (2001) 299–319, doi:10.1002/cem.681.
[43] J. Trygg, S. Wold, Pls regression on wavelet compressed nir spectra, Chemom. Intell. Lab. Syst. 42 (1–2) (1998) 209–220, doi:10.1016/S0169-7439(98)00013-6.
[44] D. Valencia, J. Salazar, J. Valencia, Comparison analysis between rigrsure, sqtwolog, heursure and minimaxi techniques using hard and soft thresholding methods, in: XXI Symposium on Signal Processing, Images and Artificial Vision (STSIVA), 2016, pp. 1–5, doi:10.1109/STSIVA.2016.7743309.
[45] X. Vang, K.K. Palival, Feature extraction and dimensionality reduction algorithms and their applications, Pattern Recognit. 30 (10) (2013) 2429–2439, doi:10.1016/S0031-3203(03)00044-X.
[46] D. Vimalajeewa, E. Robson, D.P. Berry, C. Kulatunga, Evaluation of non-linearity in mir spectroscopic data for compressed learning, in: IEEE International Conference in Data Mining, 2017.
[47] S. Wolfert, C. Verdouw, M.J. Bogaard, Big data in smart farming - a review, Agric. Syst. 153 (5) (2017) 69–80, doi:10.1145/2677046.2677052.
[48] C. Xing, L. Ma, X. Yang, Stacked denoise autoencoder based feature extraction and classification for hyperspectral images, J. Sens. 2016 (2016), doi:10.1155/2016/3632943.
[49] J. Zhang, M. Wang, Z. Li, Parallel and distributed dimensionality reduction of hyperspectral data on cloud computing architectures, IEEE Sel. Top. Appl. Earth Obs. Remote Sens. 9 (6) (2015) 2270–2278, doi:10.1109/JSTARS.2016.2542193.
[50] S. Zheng, S.R. Kulkarni, H.V. Poor, Attribute-distributed learning: models, limits, and algorithms, IEEE Trans. Signal Process. 59 (1) (2011) 386–398, doi:10.1109/TSP.2010.2088393.

Dixon Vimalajeewa is a PhD student at Telecommunications Software and Systems Group (TSSG) at Waterford Institute of Technology (WIT). His research interests include data analytics, sensor-based animal phenotypes and distributed learning algorithms (dvimalajeewa@tssg.org)

Chamil Kulatunga is a postdoctoral researcher in the Telecommunications Software and Systems Group (TSSG) at Waterford Institute of Technology (WIT). His research interests include distributed analytics, fog computing and smart agriculture (ckulatunga@tssg.org).

Donagh P. Berry is a quantitative geneticist at the Animal and Grassland Research and Innovation Centre at Teagasc. His research interests include genomic analysis, predictive modelling, chemometrics, breeding objectives and production indexes, decision support tools (donagh.berry@teagasc.ie).

Appendix B

Evaluation of Non-linearity in MIR Spectroscopic data for Compressed Learning

Conference Title:	High Dimensional Data Mining (HDM) Workshop, IEEE Conference on Data Mining , New Orleans, USA, (ICDM 2017)
Article Type	Regular Paper
Complete Author List	Dixon Vimalajeewa, Eric Robson, Donagh P. Berry, and Chamil Kulatunga
Status	Published: Nov. 2017

2017 IEEE International Conference on Data Mining Workshops

Evaluation of Non-linearity in MIR Spectroscopic Data for Compressed Learning

Dixon Vimalajeewa, Donagh Berry, Eric Robson, Chamil Kulatunga
Telecommunications Software and Systems Group, Waterford Institute of Technology, Waterford, Ireland
Teagasc, Animal & Grassland Research and Innovation Centre, Moorepark, Fermoy, Co. Cork, Ireland
Email: dvimalajeewa@tssg.org, Donagh.Berry@teagasc.ie, {erobson, ckulatunga}@tssg.org

***Abstract*—Mid-Infrared (MIR) spectroscopy has emerged as the most economically viable technology to determine milk values as well as to identify a set of animal phenotypes related to health, feeding, well-being and environment. However, Fourier transform-MIR spectra incurs a significant amount of redundant data. This creates critical issues such as increased learning complexity while performing Fog and Cloud based data analytics in smart farming. These issues can be resolved through data compression using unsupervisory techniques like PCA, and perform analytics in the compressed-domain i.e. without de-compressing. Compression algorithms should preserve non-linearity of MIRS data (if exists), since emerging advanced learning algorithms can improve their prediction accuracy. This study has investigated the non-linearity between the feature variables in the measurement-domain as well as in two compressed domains using standard Linear PCA and Kernel PCA. Also the non-linearity between the feature variables and the commonly used target milk quality parameters (Protein, Lactose, Fat) has been analyzed. The study evaluates the prediction accuracy using PLS and LS-SVM respectively as linear and non-linear predictive models.**

1. Introduction

Advances in pervasive computation and communication technologies with IoT systems result in rapid adoption of Fog/Edge computing based data analytics to discover near real-time insights in smart farming [1]. The opportunity of collecting and analyzing millions of high-resolution data demands distributed analytics across the resource-constrained Fog devices rather than centralizing raw data. Therefore efficient data storage, communication and processing techniques are vital [2] in Distributed Learning (DL) [6] compared to learning by centralizing data of such applications. This is not only because of scalability, but also due to significant contributions towards energy optimization [3], [12]. Instead of aggregating raw data, DL aggregates rich features from each data source to discover high quality global knowledge. The success of DL depends on the accuracy of knowledge aggregation at the same level where centralized learning could achieve. Therefore, one of the important task in DL is to prepare data in a compressed feature space that enables to maximize information extraction while minimizing computation, communication and storage resource consumption [2], [4].

Pasture-based dairy farming is one of the industries, which has distributed data sources in a large terrain and essentially requires such optimized systems to accelerate current farming strategies [7]. In smart dairy farming, farms are being adopted with the new technologies such as per-animal based milk yield and quality monitoring, sensor-based animal behaviour tracking [5] and robotic milking etc. to improve the quality and efficiency of dairy production. Among them Mid-Infrared Spectroscopic (MIRS) milk quality monitoring and its association analysis with other factors is vital for milk value analysis and for identifying associated phenotypes [8]. To apply DL on these datasets, a *Compressed Learning* (CL) approach (explain in Section 2) is commonly used to extract descriptive features from the raw data. Prior knowledge of the general characteristics of data is essential for a lossy CL approach to retain the precision of learning.

According to the literature [13], [14], [15], the linear/non-linear behaviour of data has a considerable impact on the accuracy of the final learning outcomes. The purpose of most of these studies were very generic because they were based on the fact that non-linear machine learning algorithms have better performances than linear techniques regardless of their complexity and the required computational power. However, linear approaches could achieve the same precision as non-linear techniques with lesser computation. However, recent data analytics, which are capable of doing complex learning with modern computational power, pay attention to employ the most accurate learning approach. Therefore, understanding the original characteristics of the data in particular, non-linearity in CL is vital.

In this study, we investigated the linear and non-linear behaviours of MIRS dataset (Fig. 1) in the context of milk quality predictions. First, pre-processing removed the impact of water absorbances from our dataset. Then non-linearity between the features in measurement-domain as well as in the compressed-domain were investigated for different milk quality parameters. Then the CL approach was used to perform learning from the compressed data, which reduced learning complexity. The impact of non-linearity were taken into account during the data compression based on linear (standard) principal component analysis (LPCA) and Kernel PCA (KPCA) techniques. The learning accuracy of using compressed-domain data was explored with a linear and a non-linear statistical predictive models; partial least square (PLS) and least squares support vector machine (LS-SVM). Section 1 has provided an introduction to the paper with its

2375-9259/17 $31.00 © 2017 IEEE
DOI 10.1109/ICDMW.2017.77

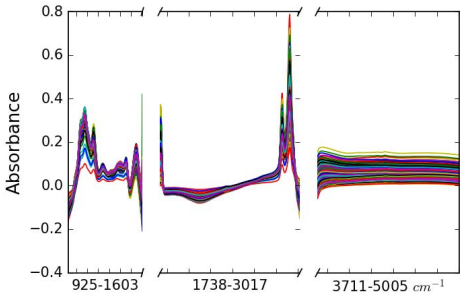


Fig. 1. Water-removed MIR spectra (X) of 712 milk samples in the wave region $925 - 5005cm^{-1}$. Water removal pre-process has reduced the feature-space dimensionality from 1060 wavenumbers to 847.

motivation. The remainder of the paper has been structured as follows. Section 2 discusses the significance of non-linearity in MIRS data and its importance in CL. Section 3 provides the methodologies we used to analyze non-linearity in MIRS milk quality predictions. Section 4 provides the analytical results based on our MIRS data followed by the conclusions in the Section 5.

2. Non-linearity and CL in MIRS

The main objective of the traditional data compression techniques is to reduce data storage and communication requirements as much as possible while minimizing information losses. These compression techniques do not contribute much to reduce learning complexity as de-compression was performed to a similar complexity prior to the learning process. The challenge of performing efficient data analytics with higher dimensions with highly redundant data remains unchanged. CL concept can be used to overcome this issue in Fog/Edge computing and in big data analytics. In CL, the original data (measurement-domain) is compressed while preserving the original learning accuracy. De-compression can be postponed until only if it is necessary. Thus, CL significantly reduces learning complexity. The data reduction techniques such as PCA and Wavelet Transformation (WT) has been used for CL [21].

Fig. 2 illustrates the CL process with unsupervised PCA compression where the measurement-domain and the compressed-domain data can either be in the same or different processing entities. Suppose matrix $Y_{n\times p}$ contains data for p response variables and need to build a regression model for those in Y (e.g. Lactose, Protein, Fat milk quality parameters) using $X_{n\times m}$ with m feature variables and n data samples. In order to preserve the original information and improve the learning performance in the compressed-domain, (l, G, P) from the compression should represent the original characteristics of X as much as possible.

In general, CL can be performed either in a single processing entity or in many geo-distributed processing entities. In a single processing scenario, both compression and learning can be supervisory since the compression unit is aware of what the compressed data is used for. Therefore, an optimal compression can be performed and continue to the learning process. In distributed scenario, compression and learning may

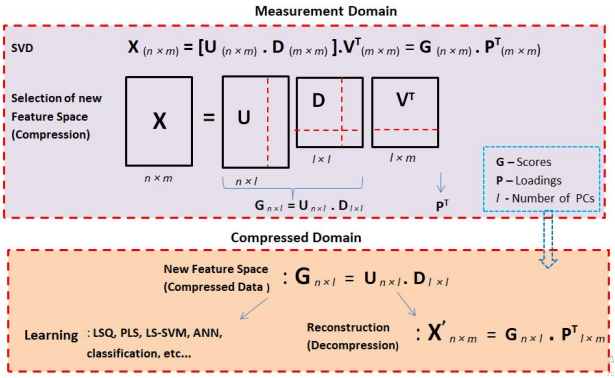


Fig. 2. The matrices of SVD used in classical PCA for deriving our compressed-domain data, which later be used in CL to predict MQPs.

be performed independently at two different locations. Unsupervised compression and a supervised learning should be employed since the compression unit may not know what will be the learning purpose. Therefore the compression entity will not aware of the most specific and relevant information required for the learning. It may neglect generally the least significant information according to the properties of the compression algorithm (e.g. variants in PCA). In order to achieve a robust analytic outcome by extracting the most accurate information, a proper understanding of X with Y is important because it helps to form well-represented compressed data and then perform learning in CL [24].

Performing a comprehensive pre-analysis with careful attention at all possible characteristics such as non-linearity, redundancy (including co-linearity), scaling and normalization of data helps to understand the data before applying CL. Therefore, such analysis overcomes the most decisive challenge in CL to select suitable compression and learning techniques based on the underlying behaviour of data. Understanding non-linearity between the feature variables and with the response variables can make a significant impact on the accuracy of CL. If a linear compression technique is used on the dataset without knowing that data has non-linear behaviours, compression may loose non-linear property of the original data. The information losses can be minimized by first understanding the behaviour though a pre-analysis and using a non-linear compression to preserve both linear and non-linear characteristics.

Most of the past studies in MIR spectrometry have followed the centralized analytics. *Y. M. Chen et al.* [13] studied non-invasive determination of sorghum species with different dimensionality reduction techniques and non-linear predictive models. Their study proved that the concern about non-linearity of MIRS sorghum data contributed for dimensionality reduction as well as improving the robustness of learning outcomes. The authors in [14] also considered non-linear associations between melamine content and MIRS spectra of dairy products (liquid milk, milk powder and infant formula). The generalization performance of linear and non-linear dimensionality reduction with a non-linear learning technique (SVM) has been studied by *L. J.*

Cao et al. [15]. They explored non-linear dimensionality reduction methods (KPCA and ICA) to capture higher order information of the input signal than linear methods (PCA). As a result, they were able to improve the generalization performance of their predictive models. In this study, we looked at non-linearity of MIRS data used in DL scenarios using CL.

3. Evaluation Methodologies

In this study, we first analyzed linear and non-linear associations between the measurement-domain variables in X and then between each compressed-domain significant feature variables (G) for three selected target variables Y (Protein, Fat and Lactose). The linear/non-linear correlation coefficients, PCA reconstruction error and non-linearity rate (NLR) measures were used with unsupervised CL (only needed X). Partial residual plots (PRP) and *Durbin-Wotsan* (DW) test were used with supervised approach (needed both X and Y) to describe the impact of non-linearity using LPCA and KPCA. PLS and LS-SVM learning approaches were used to examine the quality of compression based on non-linearity of the compressed data.

3.1. Linear/Non-linearity Evaluation Measures

Correlation Coefficients: There are different types of correlation measures such as *Pearson's correlation* (*cor*) and *Maximal correlation* (*mcor*), which are used for different purposes. *cor* captures only the linear correlation between random variables (generally called as the correlation coefficient), which is a statistical measure used to quantify association between random variables $X_i, X_j \in \mathbb{R}$,

$$cor(X_i, X_j) = \frac{cov(X_i, X_j)}{\sqrt{var(X_i)}\sqrt{var(X_j)}} \tag{1}$$

$cor(X_i, X_j) = 0$ does not mean that there is no association because *cor* cannot detect if there is a non-linear association. *mcor* enables measuring non-linear correlations by transforming the data, where associations are not detectable in the original data space and is defined as;

$$mcor(X_i, X_j) = max_{f,g} cor(f(X_i), g(X_j)) \geq 0 \tag{2}$$

where $f, g \in \mathbb{R} \rightarrow \mathbb{R}$ are two functions selected so that they maximize the correlation of X_i and X_j. If there are non-linear associations, $mcor \geq cor$ and otherwise $mcor = |cor|$. The Alternating Conditional Expectation (ACE) algorithm was used to compute *mcor* in our evaluations. In this study, *cor* and *mcor* measures [16] were used to recognize linear and non-linear associations in our MIRS data X.

NLR: NLR is a quantitative measure for the degree of non-linearity in data. Most of the measures of non-linearity are based on the residuals from linear and non-linear regression fittings. The residual difference between two fittings gives an idea about the non-linearity. According to [11], NLR can be defined assuming that non-linear techniques fit perfectly to the data (i.e. non-linear fitting residual error is nearly zero).

$$\begin{aligned} NLR &= \frac{1}{n\sigma}\sum_{i=1}^{n}(||L_i - X_i||^2 - ||H_i - X_i||^2) \\ &\simeq \frac{1}{n\sigma}\sum_{i=1}^{n}||L_i - X_i||^2 \end{aligned} \tag{3}$$

where n number of data points (X_i), L_i and H_i are supporting points of linear and non-linear regression fittings, respectively. $\sigma = \frac{1}{n}\sum_{i=1}^{n}||X_i - \mu||^2$ is the variance of data X and μ is the mean of X. The Equation (3) indicates the amount of residuals from linear fitting. Higher NLR will result in higher non-linearity and vise-versa. Suppose the linear fitting is LPCA, then NLR can be derived as follows.

$$NLR = 1 - \frac{\sum_{i=1}^{l}\lambda_i}{\sum_{i=1}^{m}\lambda_i} \tag{4}$$

where λ is the i^{th} eigenvalue computed form the covariance matrix of X and l is the selected dimension for $l = 1, 2, \cdots, m$. The proof of this formula can be found in [11].

PRP: A plot obtained using Least Squares Regression (LSR) fitting can be used to understand the usefulness of the LSR model parameters and their unknown functional forms (e.g. non-linearity). According to [23], partial residuals (component+residuals) are the residuals of a LSR model fitting added to the mis-specified part of the model. PRPs are the plots of partial residuals against the mis-specified part. Suppose a LSR model in the form;

$$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_m x_m + f(x_k) + \epsilon \tag{5}$$

where $f(x_k), k = 1, \cdots, m$ is an unknown function to be identified (mis-specified part), β_i's are the LSR model parameters of the predictor variable x_i's for $i = 1, \cdots, m$ (β_0 is intercept) and ϵ is the random error. PRP of f gives an graphical overview regarding the effect of f to y when the effect of all other x_is are controlled. This concept was used to check the relationship of each feature to their corresponding response.

DW Test: This statistical test is used as a measure of auto-correlation (ρ) of residuals from a LSR fitting to check whether there is a correlation between the successive residuals. Since, residual ρ indicates the goodness of LSR fit, this can be used as a technique to identify the relationship (linear/non-linear) of response variables to its feature variables. The null hypothesis states $H_0 : \rho = 0$ and alternative hypothesis states $H_1 : \rho > 0$. The test statistic d is computed by,

$$d = \frac{\sum_{i=1}^{n-1}(e_i - e_{i-1})^2}{\sum_{i=1}^{n}e_i^2} \tag{6}$$

where e and n reflect a residual and the number of samples. If $d < d_L$, H_0 is accepted (residuals are uncorrelated and normality exists in the model). H_0 is

rejected, if $d > d_U$, which reflects that there exists a correlation in residuals and linearity in the model. The test is inconclusive, if $d_L < d < d_U$. The d_L and d_U are lower and upper critical values for the test [19].

3.2. Data Compression and Regression Methods

PCA is used for dimensionality reduction, visualizations, compression (with loss), de-noising (removing small variance in the data) and whitening (de-correlation so that features have unit covariance). PCA is a variance-based statistical dimensionality reduction technique. It draws a low dimensional space and represents each data point by its projection along the orthogonal directions, which represents maximal variance of the data. The low dimensional space is called compressed-domain feature space and the projections along the directions are called principal components (PCs). We used LPCA and KPCA [10] in our evaluations of CL.

LPCA: Fig. 2 shows the process of LPCA with singular value decomposition (SVD) within the measurement-domain entity. Given the mean-centered data set ($X_{n\times m}$), SVD decomposes X in the form $X = UDV^T$, where U, V are respectively upper and lower triangular matrices where $U^TU = I = V^TV$ (I is an identity matrix). D is a diagonal matrix in which elements follow the condition $d_{11} \leq d_{22} \leq \cdots \leq d_{mm}$ (eigen values of the co-variance matrix of X). Then the score matrix $G = UD$ (compressed feature space) and the loading matrix $P = V$ are derived. Finally, the data is transformed into its compressed-domain by selecting the scores of the significant l ($< m$) PCs, which minimizes the reconstruction error. LPCA is based on the assumption that correlations are linear.

KPCA: When data has complex non-linear associations, which is more realistic in practical datasets such as MIRS data, KPCA like non-linear feature extraction methods have to be used for data compression in CL [10], [15], [20]. It has been proved in many studies that non-linear methods perform well in dimensionality reduction by capturing global characteristics in data [15]. In KPCA, the original data matrix $X_{n\times m} \in \mathbb{R}^m$ is mapped into a new higher dimensional space (feature space) $\mathbb{F}^M$ by a non-linear function ϕ such that,

$$\phi \colon \mathbb{R}^m \rightarrow \mathbb{F}^M \tag{7}$$

For a certain selection of ϕ, $\mathbb{F}^M$ has arbitrarily large dimension and then LPCA is performed using "*kernel trick*". According to the *Mercer*'s theorem, non-linear mapping function ϕ and the kernel function K are associated by the equation $K(x^i, x^j) = \phi(x^i)^T \cdot \phi(x^j)$. Given the kernel function K, the normalized kernel matrix $\hat{K}_{m\times m}$ of the data $X_{n\times m}$ is computed as follows.

$$\hat{K} = K - 2I_{1/n}K + I_{1/n}KI_{1/n} \tag{8}$$

where $I_{1/n}$ is a matrix with all elements $1/n$. Then LPCA is applied on $\hat{K}$ in the feature space, which is equivalent to non-linear PCA in the original data domain. There are different types of kernel functions such as Gaussian (radial basis - RBF) and polynomial [15] where the RBF kernel; $K(x, x_i) = exp(-||x - x_i||^2/\sigma^2)$ was used with $\sigma^2 = 0.25 \times m \times mean(var(X))$ in our MIRS data compression.

PLS: The ordinary LSR derives a relationship between X and Y with the assumption that X variables are uncorrelated. However, since some data such as MIRS data violates this assumption, PLS builds regression models by considering correlations of variables in X itself as well as between X and Y. Therefore PLS is considered as a bilinear modelling method in which X data is projected into a feature space (or latent variables, LVs) and then simplify relationship between X and Y to predict Y selecting least number of LVs via cross-validation. First, decompose both X and Y as the decomposition was performed in LPCA;

$$X = TP^T + H \qquad Y = RQ^T + L \tag{9}$$

where, T and R are the score matrices and P and Q are the loading matrices. H and L are respectively the error matrices, which come from the process of PLS regression of X and Y. Then, LSR is applied for scores T and R such that $R = WT + e$, where W and e are respectively the weight matrix (to be estimated) and the error term, which fits a LSR model for X and Y [9].

LS-SVM: As same as the process explained in KPCA, when data has complex non-linear associations, linear models cannot capture them properly. Therefore, LS-SVM is used to form a regression model in the feature space $\{\phi(x^i)\}_{i=1}^n$. The regression model in LS-SVM is given by,

$$y(x) = W^T\phi(x) + b \tag{10}$$

where $W \in \mathbb{R}^n$ is the weight vector and b is the bias. LS-SVM is an optimized algorithm based on the standard SVM [18]. The optimization problem is formulated as follows.

$$minJ(W, e) = \frac{1}{2}W^TW + \frac{1}{2}\gamma\sum_{i=1}^{n} e_i^2$$

where γ is the regularization parameter and e_i is the random error. The *Lagrange multiplier* method is used to solve the optimization task in the LS-SVM algorithm.

$$L(W, b, e, \alpha) = J(W, e) - \sum_{i=1}^{n} \alpha_i\{W^T\phi(x_i) + b + e_k - y_k\} \tag{11}$$

where α_i is Lagrange multipliers. The above Equation (11) is solved by partial differentiation with respect to each variable. Then estimation function of y can be obtained as,

$$y(x) = \sum_{i=1}^{n} \alpha_k K(x, x_k) + b; \quad i, j = 1, 2, 3, \cdots, n \tag{12}$$

where K is the kernel function. The selection of the parameter values γ and σ (RBF kernel parameter) is important. This is because γ improves the generalization performance of the model and σ controls the regression error and also reflects the sensitivity of LS-SVM model due to noise in input variables [17]. Thus, large γ and σ reflect respectively more non-linear model and global properties. There are different techniques to set parameter values in LS-SVM model such as cross-validation, grid search, Bayesian optimizer [22].

4. Evaluation Results

4.1. MIR Spectroscopic Milk Quality Data

The data used in this paper has been obtained from Teagasc research dairy farm at Moorepark, Ireland where MIR spectra was collected (in 35 days starting from August 2013 and ending in August 2014) using 605 different dairy cattle. The composition of milk was determined using FOSS MilkScan prediction equations using FT-MIR technology. The input data matrix contained the spectra of 712 different milk samples in the wavenumber region $925 - 5005cm^{-1}$ with a resolution of $3.853cm^{-1}$. When the wavenumbers were rounded to the nearest integer, a given spectrum contained 1060 transmittance data points. Therefore, the original MIRS spectra used (called the gold standard) to apply compression algorithms was a (712×1060) dimensional matrix.

Since spectral values were given in transmittance, we converted them to absorbance by taking log_{10} of the reciprocal of given transmittance values. According to the impact of water absorption in MIRS at $25°C$, two corresponding wave regions were removed as $1607 - 1734cm^{-1}$ and $3021 - 3707cm^{-1}$. This reduced our spectra to 847 wavenumbers, which we used as the input data matrix X (Fig. 1) in our analysis. In addition, the percentages of the selected MQPs corresponding to each sample were stored in a matrix (Y). Among them three most commonly used MQPs; Lactose, Protein and Fat were taken into the evaluations. Then our data compression and regression model calibration/validation were applied on this gold standard data. *R*-software was used for non-linearity analysis and *MATLAB* was used for PLS and LS-SVM model building and evaluations.

4.2. Non-linearity in Measurement-domain Data

To emphasize that there are linear and non-linear correlations in X, the *cor* and *mcor* were computed for all every pairs of wavenumbers in X. The results are shown in Fig. 3 with their absolute differences. According to the variations of color intensity, there are high and low variations respectively in the regions 925-3025cm^{-1} and 3025-5005cm^{-1}. The figure shows both linear and non-linear correlations in the region 925-3025cm^{-1} due to strong *mcor*s values. Even though $mcor \geq cor$ among the wavenumbers in both regions 925-3025cm^{-1} and 3025-5005cm^{-1}, the region 925-3025cm^{-1} shows a higher variation. Both correlations seem to be similar (linear/no correlation) among the

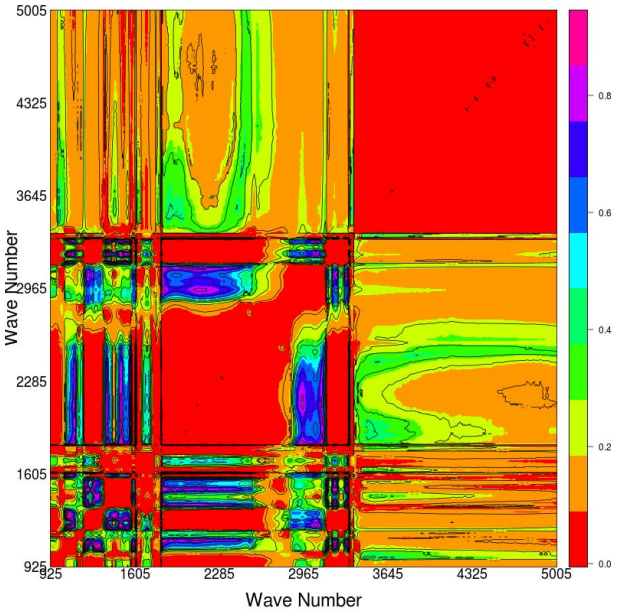


Fig. 3. The absolute difference of Maximal (*mcor*) and Pearson's (*cor*) correlation coefficients ($|mcor - cor|$) between feature variables in X.

wavenumbers in the region 3025-5005cm^{-1} and some in 952-3025cm^{-1} since there are no much color variations (red regions). Within the region of 925-3025cm^{-1}, at some points the correlation difference is even greater than 0.6. Maximum of 0.96 was observed between the wavenumbers $1387cm^{-1}$ and $1152cm^{-1}$ (the strongest *mcor*).

In general, there are both linear and non-linear associations in X and in particular, more non-linear correlations exist among the wavenumbers up to the $3025cm^{-1}$. The correlations between the wavenumbers above $3025cm^{-1}$ are not much stronger in terms of non-linearity.

To explore the importance of the linear/non-linear correlations of the feature variables in X, PLS regression was applied and the regression coefficients (βs) were derived for each MQP. Then the correlations of the wavenumber at significant βs (e.g. $\beta \geq 3\sigma_\beta$) with the other features were computed for each MQP. Fig. 4 shows the βs of each MQP and the absolute correlations (*mcor*s and *cor*s) of wevenumbers at the highest significant β (Lactose - $1745cm^{-1}$, Fat - $1734cm^{-1}$ and Protein - $1541cm^{-1}$) with other coefficients.

In each plot, $mcor \geq cor$ for all βs and most of the correlations are high and fluctuated sharply within the region 925-3012cm^{-1} compared to the correlations of the βs above the wavenumber $3710cm^{-1}$. The plot for Protein are clearly non-linear because the differences between *mcor* and *cor* are higher for many βs. Even though there is no much non-linearity in the plots for Lactose and Fat compared to Protein, correlations in the region 2730-2817cm^{-1} show a clear non-linearity. Thus these plots reveal that the correlations associated with the most significant β are non-linear for Protein compared to the correlations associated with the most

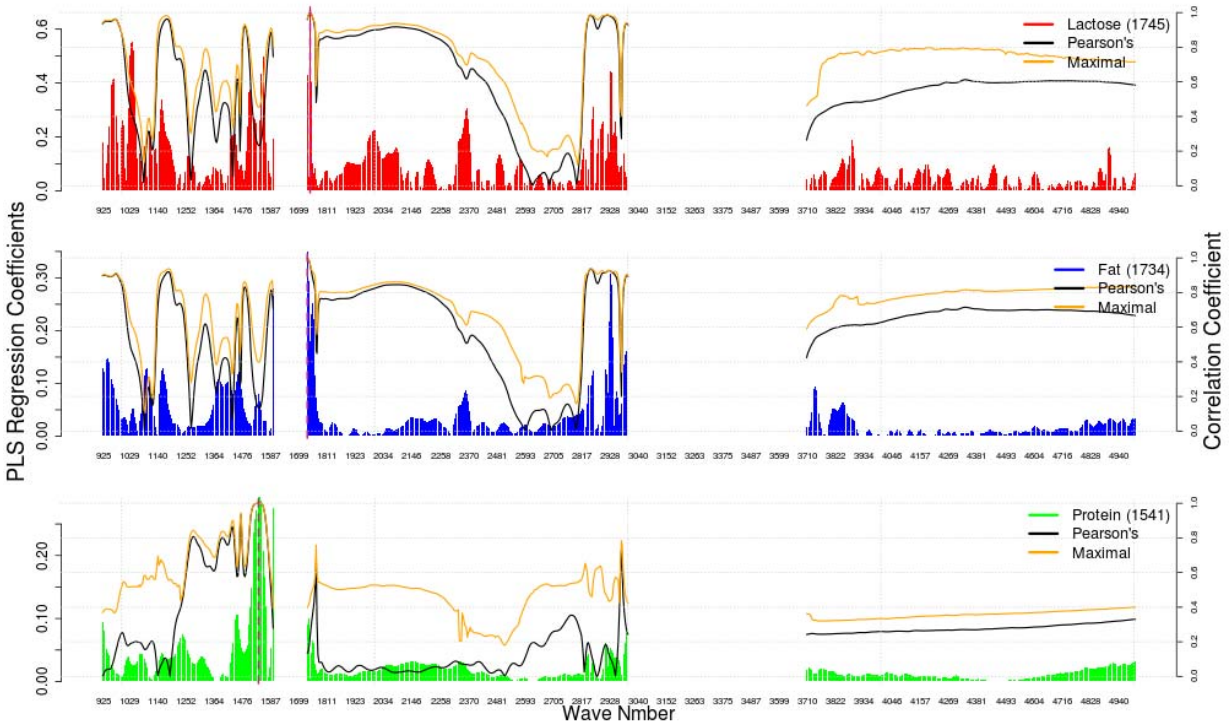


Fig. 4. PLS regression coefficients and the absolute values of *mcor* and *cor* correlations for Lactose, Fat and Protein. The shown correlation coefficients are between the wavenumber at the most significant regression coefficient with the rest of the wavenumbers.

significant βs of Lactose and Fat.

With regard to βs of each MQP, the highest significant βs lie in the regions where both *mcor* and *cor* are more or less similar (linear correlation) in each plot. There are many coefficients those lie where the correlations are non-linear. For instance, correlations of βs of Protein in the regions 925-1250cm^{-1} and 2034-2370cm^{-1} contain most of the significant coefficients. If data is compressed with LPCA, then the wavenumbers in these regions will likely to be removed due to lesser *cor* correlations. As a consequence, a high information loss can happen. Non-linear KPCA may be able to capture those non-linear as well as linear associations. Therefore, KPCA compressed data may retain more characteristics from the original data than LPCA. Therefore, it is important to understand the non-linearity as a prior knowledge before applying CL.

4.3. Performance of Linear/Non-linear PCA

After analyzing the behaviour of correlations in the dataset X, its impact on the PCA compression was investigated. The amount of information captured by LPCA and KPCA algorithms were considered by computing REs for the first 100 PCs. The results are shown in Fig. 5 (left). According to the figure, REs of KPCA is less than that of LPCA. It turns out that KPCA incurs lesser REs with a lower number of PCs than in the LPCA. According to KPCA, this is due to existence of non-linearity in X. For instance, REs of LPCA and KPCA with 20 PCs are respectively 5.9×10^{-4} and 8.6×10^{-7}. Therefore, LPCA needs more PCs to achieve

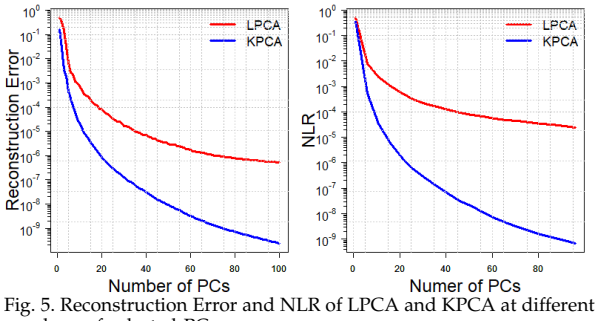


Fig. 5. Reconstruction Error and NLR of LPCA and KPCA at different numbers of selected PCs.

the same RE where KPCA can achieve with a lesser number of PCs. It confirms that the nature of associations among the variables in data directly affect the compression.

To further investigate the existence of non-linearity in X and its impact on compression, the degree of NLR was computed with LPCA and KPCA at different number of PCs. Fig. 5 (right) shows the variation of NLR which represents the linear fitting residual error. LPCA incurs a higher degree of NLR than KPCA, which means that non-linear fitting produces lower residual errors than linear fitting, which confirms the outcomes of REs. For instance, NLR with 20 PCs is 2.99×10^{-4} of LPCA, which is twice higher that in KPCA (5.59×10^{-8}). It confirms that there is a non-linearity between feature variables in X. KPCA captures non-linearity better than LPCA. Further, the degree of

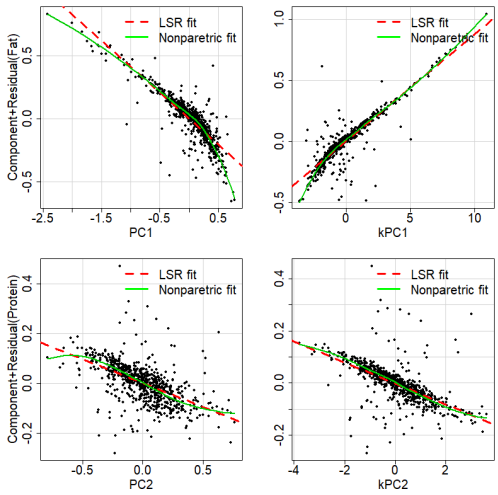


Fig. 6. PRPs for PC1 (Fat) and PC2 (Protein) to detect non-linearity in MIRS data with the first 10 PCs of LPCA (left) and KPCA (right).

NLR decreases with increasing number of PCs. This shows that extraction of higher dimensional feature space from the original data has lower degree of non-linearity.

4.4. Non-linearity between the Features and Target Variables

DW test was conducted to investigate the non-linearity between feature variables in X and the response variables based on LSR modeling. The scores of the first 10 PCs derived from LPCA and KPCA were used to compute LSR residuals to evaluate DW test statistics for each MQP. The test statistics and corresponding critical values are given in the Table 1. The results reveal that Protein and Fat predictions have non-linear behaviors and Lactose has a linear relationship in the MIRS dataset X.

To make a visual interpretation of non-linearity in Fat and Protein predictions, which was evidenced by DW test, their PRPs were drawn using the scores of the first 10 PCs. To get an idea about non-linearity of Fat and Protein, only the PRP of PC1 and PC2 are shown in Fig. 6 (LPCA-left and KPCA-right). The divergence of the non-parametric fit from the fitted LSR line indicates a degree of non-linearity and the type of non-linear relationship. The PRPs from the compressed-domain of LPCA represent higher divergence from the LSR fit than those using compressed-domain of KPCA. This confirms that Protein and Fat predictions have non-linear relationships to X.

Table 1: DW TEST for the SELECTED MQPs WITH FIRST 10PCs DERIVED FROM LPCA AND KPCA ($d_L = 1.8498$ and $d_U = 1.9019$).

MQP	DW test statistic (d)		Decision (linear/non-linear)
	LCPA	KPCA	
Lactose	1.6559	1.6391	$d < d_L$ linear
Fat	1.9680	1.9505	$d > d_U$ non-linear
Protein	1.9805	1.9779	$d > d_U$ non-linear

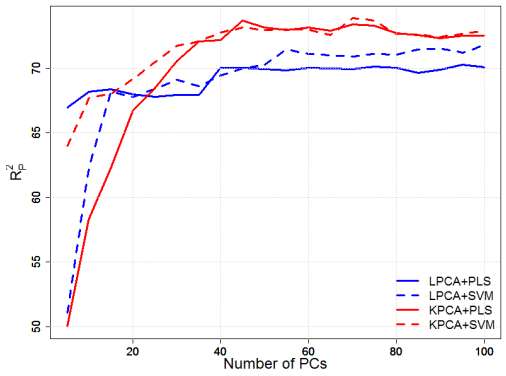


Fig. 7. Prediction accuracy of PLS and LS-SVM based on LPCA and KPCA compressed data for Protein.

4.5. Learning Accuracy of PLS and LS-SVM

To study the learning performance of the regression models based on LPCA and KPCA compressed data, the learning accuracy was computed from PLS and LS-SVM. First, the dataset was divided into two subsets as calibration (80% of the samples) and validation (the remaining samples) by using *Kernard-stone* sampling method. Calibrations and validations were performed for the first 100 PCs and computed the validation R_P^2 (coefficient of determination) as the learning accuracy. The number of latent variables (LVs) in PLS was selected employing *10-fold* cross-validation. We used the *Bayesian* optimization approach to select LS-SVM model parameters; γ and σ.

Fig. 7 shows the learning accuracy computed from PLS and LS-SVM models for Protein with different PCs derived from LPCA and KPCA. Almost the same maximum learning accuracy of nearly 74% was achieved with KPCA compressed data using first 45 and 70 PCs respectively from PLS and LS-SVM. The learning accuracy was higher with LPCA compressed data than in KPCA only for lower number of PCs (around $\leq$ 20 PCs). The number of PCs where the maximum learning accuracy was achieved is higher with KPCA than LPCA. In general, comparing all the values, it turns out that non-linear compression and learning has improved the leaning accuracy although the feature space is higher compared to the linear approach.

The same procedure was repeated for Fat and Lactose and the results are shown in Table 2. The observed highest learning accuracies from the original and LPCA/KPCA compressed data with the corresponding PLS and LS-SVM model parameters (including Protein). The number of PCs of with those accuracies were observed are also given. All the learning outcomes show that CL gives higher leaning accuracy than that was obtained from the original data. Further, the performance of CL from KPCA compression is better than that of LPCA except Lactose. The learning performances of LS-SVM models are always higher than PLS models regardless of the compression technique. The results show that the learning accuracy from LPCA compressed data is higher than KPCA. This turns out that there is a linear relationship between MIRS data for Lactose prediction.

Table 2: COMPRESSED-DOMAIN LEARNING ACCURACY (R^2) of PLS AND LS-SVM PREDICTIONS for LACTOSE,FAT AND PROTEIN

MQP	Original Data		LPCA				KPCA			
	PLS	LS-SVM	PLS		LS-SVM		PLS		LS-SVM	
	R^2_P(LVs)	$R^2_P(\sigma,\gamma)$	$\#PCs$	R^2_P	$\#PCs$	R^2_P (σ,γ)	$\#PCs$	R^2_P	$\#PCs$	R^2_P (σ,γ)
Lactose	83.51 (12)	87.59 (26.57, 58.48)	55	83.49	45	88.03 (12.25, 66.91)	95	78.13	95	79.32 (2.28,12.18)
Fat	88.04 (5)	89.35 (35.66, 69.56)	65	88.20	45	88.82 (17.56, 50.5)	55	88.17	75	88.89 (14.2, 63.5)
Protein	70.3 (15)	72.2 $(5.01, 7.4 \times 10^4)$	40	70.41	55	71.63 $(42.35, 1.58 \times 10^3)$	45	73.67	70	73.9 (39.66,64.47)

LS-SVM model parameters confirm a non-linearity for Protein and Fat predictions and a linearity for Lactose within the MIRS dataset. This is because the highest γ values were observed for Protein with both LPCA and KPCA. γ for Fat was higher than Lactose. Further, these values verify the results given in Fig. 7 and Table 1. The behaviour of σ values was same as γ, which means Protein predictions have more global behaviour than Fat and Lactose.

5. Conclusions

First we investigated non-linear behaviours between the wavenumbers (features) of the MIRS data. Our investigation has shown that there is a considerable non-linearity exists and should be captured by the compression algorithms. Then we have compressed the original data using both LPCA/KPCA and investigated non-linearity between the compressed-domain feature variables and with three selected response variables (Fat, Protein and Lactose). According to this analysis, we conclude that Fat and Protein predictions show non-linear behaviours, which we need to capture in compressed learning. Finally we applied PLS and LS-SVM regression models on the two compressed-domain data to show that there is an improvement in accuracies using non-linear predictions. Therefore, we conclude that use of a linear unsupervised compression technique has negative impacts on the prediction accuracy of different MQPs. Use of non-linear compression techniques such as KPCA at the compression entity is highly desirable in compressed learning approach. Otherwise, the advantages of using complex non-linear predictive models will not be useful in MIRS based milk quality analytics.

Acknowledgment

This research was supported by the Science Foundation Ireland (SFI) through the project "PrecisionDairy" (ID: 13/1A/1977).

References

[1] S. Wolfert, L. Ge, C. Verdouw, M. J. Bogaard, Big Data in Smart Farming - A Review, Elsevier J. Agricultural Systems, vol. 153, May 2017.
[2] B. H. Park, H. Kargupta, Distributed Data Mining: Algorithms, Systems and Applications, Data Mining Handbook (Editor: Nong Ye), 2002.
[3] F. Jalali, et al., Fog Computing may help to save Energy in Cloud Computing, IEEE Journal on Selected Areas in Communications, May 2016.
[4] H. Zheng, S. R. Kulkarni, H. W. Poor, Dimensionally Distributed Learning Models and Algorithms, IEEE International Conference on Information Fusion, July 2008.
[5] K. Bhargava, S. Ivanov, C. Kulatunga and W. Donnelly, "Fog-enabled WSN system for animal behavior analysis in precision dairy", IEEE International Conference on Computing, Networking and Communications (ICNC), Jan. 2017.
[6] M. Chen, et al., On the Computation Offloading at Ad Hoc Cloudlet: Architecture and Service Modes, IEEE Communications Magazine, vol. 53 (6), June 2015.
[7] C. Kulatunga, L. Shalloo, W. Donnelly, E. Robson, S. Ivanov, Opportunistic Wireless Networking for Smart Dairy Farming, IEEE IT Professional Magazine, vol. 19 (2), March 2017.
[8] G. Visentin, et al., Prediction of Bovine Milk Technological Traits from Mid-Infrared Spectroscopy Analysis in Dairy Cows, J. Dairy Science, vol. 98, September 2015.
[9] P. H. Garthwaite, An Interpretation of Partial Least Squares, J. American Statistical Association, vol. 89, March 1994.
[10] L. J. P. ven der Maaten, In Introduction to Dimensionality Reduction using MatLab, MICC report, July 2007.
[11] W. Huang, H. Yin, Linear and Nonlinear Dimensionality Reduction for Face Recognition, IEEE International Conference on Image Processing (ICIP), 2009.
[12] Mu Li, et al., Scaling Distributed Learning with the Parameter Server, USENIX Operating Systems Design and Implementation (OSDI), 2014.
[13] Y. M. Chen, et al., Combination of the Manifold Dimensionality Reduction Methods with Least Square Support Vector Machines for Classifying Species of Sorghum, Scientific Reports, vol. 6, Jan 2016.
[14] R. M. Balabin, S. V. Smirnov, Melamine Detection by MIRS and NIRS: A Quick and Sensitive Method for Dairy Products Analysis Including Liquid Milk, Infant Formula, and Milk Powder, J. Talanta, 2011.
[15] L.J. Cao, et al., A comparison of PCA, KPCA, and ICA for Dimensionality Reduction in Support Vector Machine, J. Neurocomputing, 2003.
[16] H. V. Nguyen et. al., Multivariate Maximal Correlation Analysis, International Conference on Machine Learning, 2014.
[17] Yan-de Liu, et al., On-Line Predicting Soluble Solids Contents of Intact Pears Combination with Wavelet Transform and Support Vector Regression, IEEE International Conference on Natural Computation, 2010
[18] T. V. Gestel, et al., Benchmarking Least Squares Support Vector Machine Classiers, J. of Machine Learning, vol. 54, 2004.
[19] J. V. Anand, S. Titus, Regression based Analysis of Effective Hydrocast in Underwater Environment, IEEE Region 10 Conference on TENCON, 2014.
[20] G. A. Licciardi, F. D. Frate, Pixel Unmixing in Hyperspectral Data by Means of Neural Networks, IEEE Tran. on Geoscience and Remote Sensing, vol. 49, 2011.
[21] R. Calrebank, S. Jafarpor, R. Schapier, Compressed Learning: Universal Sparse Dimensionality Reduction and Learning in the Measurement Domain, 2009.
[22] K. Pelckmans, et al.,LS-SVMlab: A MatLab/C Toolbox for Least Squares Support Vector Machines, (http://www.esat.kuleuven.be/sista/lssvmlab)
[23] J. W. McKean, Exploring Data Sets using Partial Residual Plots based on Robust Fits, IMS Lecture Notes-Monograph Series, vol. 31, 1997.
[24] Y. Bengio, A. Courville, P. Vincent, Representative Learning: A Review and New Perspectives, IEEE Tran. on Pattern Analysis and Machine Intelligence, vol. 35, Aug. 2013.

Appendix C

Leveraging Social Network Analysis for Evaluating Animal Cohesion

Journal Title:	IEEE Transactions on Computational Social Systems
Article Type	Regular Paper
Complete Author List	Dixon Vimalajeewa, Sasitharan Balasubramaniam, Bernadette O'Brien, Chamil Kulatunga, and Donagh P. Berry
Status	Published: vol. 6, no. 2 pp. 323-337, Mar 2019

Leveraging Social Network Analysis for Characterizing Cohesion of Human-Managed Animals

Dixon Vimalajeewa, Sasitharan Balasubramaniam, Bernadette O'Brien, Chamil Kulatunga, and Donagh P. Berry

***Abstract*— Social network analysis (SNA) is a technique to study behavioral dynamics within a social group. In SNA, it is an open question whether it is possible to characterize animal-level behaviors by using group-level information. Also, it was believed that the combined use of SNA would provide a more comprehensive understanding of social dynamics. In light of these two factors, here we explain an approach to evaluate animal importance to a group by considering the variability in group-level structural information, which is computed by joining the animal- and group-level SNA measures node centrality and network entropy, respectively. Moreover, two other metrics, animal social interaction range and nearest-neighbor frequency matrix, which represent a social affiliation of each animal within the group, are computed to help address the general challenges in graph-based SNA and, thereby, improve the precision of animal importance measures. Finally, we derive the joint distribution of animal importance of the group in detecting atypical social behaviors. The approach is tested using tracking data of dairy cows. The reliability of the derived animal importance was superior to the already existing animal importance measures. To illustrate the usability of the animal importance metric, a simulation study was conducted to identify sick and estrus animals in a group. The social affiliation of sick cows was less when compared to healthy cows. Also, their individual distributions of animal importance were shifted toward the left of the mean of the animal importance distributions of healthy cows. Consequently, the joint distribution of animal importance of the group exhibited a bimodal distribution with a left tailored shape. The behavior of cows in estrus was opposite to that of sick cows. Moreover, with the increasing number of sick and estrus cows in the group, respectively, the group entropy decreased with larger variance and slightly increased with less variance. Therefore, the entropy-based animal importance metric has superior performances when evaluating animal importance to the group compared to the existing metrics. It can be used for generating alerts for the early detection of atypical social behaviors associated with, for instance, animal health, veterinary, and welfare.**

***Index Terms*— Animal importance, social affiliation, social interaction range, social network analysis (SNA).**

I. Introduction

FARM animals are usually gregarious animals and their social interactions could greatly be capitalized on improving farm management operations such as improving animal well-being [1]. Evidence that supports the significance of their social relationships is considerably underutilized [2]. The growing interest in social network analysis (SNA), which is a popular method to study the behaviors social groups, confirms that it can offer great advantages by exploring individual social affiliations [3]. Also, recent advancement in animal monitoring systems which enable the collection of vast amount of data, such as wireless sensor networks (WSNs), facilitates large-scale SNA in broader perspectives [4]. Consequently, this necessitates the investigation of novel strategies for SNA to transform animal social relationships into useful metrics, which can subsequently be used to support better farm management practices.

Modern-day animal production systems routinely exploit state-of-the-art technologies to support decision making by generating valuable insights from data collected by sensors. Most pervasive sensor technology is the global positioning system (GPS), which determines geolocation [4], and therefore, the movements of individual instrumented animals. Other energy-efficient techniques such as Wi-Fi, bluetooth, and Long Range wireless signal strength-based positioning technologies are also becoming increasingly popular [5], [6]. Therefore, opportunities exist to define novel and informative insights (e.g., phenotypes) from such geolocation data. However, the necessary descriptive metrics need to be defined [7] from which deviations can be identified and subsequent alerts generated. Such metrics should be informative to characterize not only the animal itself but also the group or subgroup level behaviors. Once the behaviors of an individual animal are characterized relative to the group, any deviations from the norm can be used as early and real-time alerts for producers

Manuscript received September 23, 2018; revised January 13, 2019; accepted February 25, 2019. Date of publication March 20, 2019; date of current version April 1, 2019. This work was supported in part by the Science Foundation Ireland through the PrecisionDairy Project under Grant 13/1A/1977, in part by the Department of Agriculture, Food and Marine, Government of Ireland under Grant 16/RC/3835 (VistaMilk), and in part by the Horizon 2020 GenTORE Project. *(Corresponding author: Dixon Vimalajeewa.)*

D. Vimalajeewa is with the Telecommunications Software and Systems Group, Waterford Institute of Technology, Waterford, X91 P20H Ireland (e-mail: dvimalajeewa@tssg.org).

S. Balasubramaniam is with the Telecommunications Software and Systems Group, Waterford Institute of Technology, Waterford, X91 P20H Ireland, and also with the Faculty of Information Technology and Communication Sciences, Tampere University, FI-33014 Tampere, Finland (e-mail: sasib@tssg.org).

B. O'Brien and D. P. Berry are with Teagasc, Animal and Grassland Research and Innovation Centre, Cork, P61 C996 Ireland (e-mail: bernadette.o'brien@teagasc.ie; donagh.berry@teagasc.ie).

C. Kulatunga is with the University College Dublin, Dublin 4, Ireland (e-mail: chamilkul@gmail.com).

Digital Object Identifier 10.1109/TCSS.2019.2902456

to identify, for instance, atypical behaviors of animals that may require particular attention.

Animals living in groups are generally social animals and have complex relationships and social behaviors [8]. For instance, cows are social animals [9] and prefer to stay in groups, synchronizing their activities mostly based on neighboring (or dominant) animals. Although external stimuli can perturb normal group behaviors of such animals, there is a tendency for the animals to revert to normal status as soon as possible [10]. This is because of animal cohesion, which is a phenomenon that describes how strongly animals are connected to each other and is used to understand behavioral dynamics based on their heterogeneous social relationships in the group [9]. Changing such dynamics is mostly influenced by a small set of group members generally identified as the most important (or highly cohesive/interactive) animals to the group [11]. Hence, understanding the individual's importance to the group would be useful for exploring different behaviors of groups as well as individuals such as group synchronization, spreading dynamics, and cascading reactions [12].

SNA provides a comprehensive understanding of social dynamics among animals and also facilitates quantifying them [12], [13]. Graph theoretical concepts are commonly used for SNA. A group is depicted as a graph in which nodes represent individual animals and an edge represents the interaction between two animals (Fig. 1). This model is, therefore, capable of representing the complex structure of interactions among individuals, also called a sociogram [12]. The heterogeneous nature of interactions is then used to determine the importance of each animal. The importance of an animal to the group is recognized as its involvement in forming a complex network structure, i.e., to what extent that animal is connected to the others in the network, and quantified as the centrality. Importance is generally derived by exploiting an individual's local level structural information (only within the social interaction range) such as degree (i.e., the number of direct edges on each node) and closeness (i.e., the shortest path length between a particular edge with all other edges) [12]. Therefore, the temporal evolution of centrality of group living animals is considered as one of the relevant characteristics for defining metrics which can subsequently be used to describe different behavioral patterns in SNA [14].

Different ideas have been highlighted in order to assist in developing novel metrics for SNA. Among those ideas, Wey *et al.* [12] stated that the combined use of local, intermediate, and global measures would provide a more comprehensive understanding of the network dynamics in a broader view, based on the level of information used to quantify different social characteristics where graph measures are divided into three categories as local, intermediate, and global (or graph). In addition, Qi *et al.* [15] unveiled that using a broader range of information in quantifying network properties would provide superior performances than being limited to the domain that they have been defined for. Therefore, it is an open question whether it is possible to derive local (animal)-level measures based on the global (group)-level structural variability of social dynamics [16]. Therefore, this paper is an attempt to apply all these concepts together in order to derive a metric to evaluate animal importance (or sociality) to a group (AIm) based on the dairy cattle mobility observations. An additional aim is to explore the possible opportunities, which can be benefited by using the derived AIm, in relation to the dairy cattle behaviors as a use case.

There are different challenges, however, which cannot be disregarded when deriving novel measures, such as spatial and temporal limitations, defining the network and sampling animals into it, and validity and robustness of derived network measures [12]. Therefore, prior to conducting an SNA, careful attention should be given to those factors and the necessary actions should be in place to mitigate against them. Although the graph-based SNA generally assumes that the interactions included in the graph are relatively stable over time, which is not always true because animal relationships are highly dynamic and so will the network topology. Also, sampling relevant animals into the network, including their significant interactions, is vital as if there are too many animals, the network might split into unconnected subgroups. This may cause complications in applying graph theoretic measures such as diameter, which could consequently lead to misleading information. Practically, animal interactions depend on various factors such as age, gender, and health, so that selecting only the significant interactions is crucial [11]. Disregarding such interactions could negatively impact on the representativeness of the network and so will the robustness of the derived measures. Consequently, not only would the reliability of the estimated network dynamics become invalid but also would contribute to misleading outcomes about social behavioral dynamics.

Therefore, the main contribution of this paper is to derive an approach to evaluate AIm by combining local- and global-level measures, node centrality, and network entropy addressing the SNA challenges mentioned above. While network entropy depicts the amount of information encoded within a network and is used to compute the structural complexity at the group level [17], the node centrality quantifies node importance based on the extent to what a node is surrounded by other nodes [18]. In this approach, the influence of an animal on changing the network entropy is considered as its AIm. The idea is to taking into account the variability of graph-level structural information in evaluating node-level properties. Thus, this approach facilitates expanding the range of information used in quantifying AIm. In addition, this process is backed by two other metrics, animal interaction range and nearest-neighbor (NN) frequency matrix. The animal interaction range, which stands for the optimum range where an animal can make strong interactions, helps to sample significant interactions into the network graph. Therefore, it helps to improve the representativeness of the network graph and to enhance the validity of AIm. The NN frequency matrix demonstrates the social affiliation of each animal with others in the group as a frequency value counted over the time. As this matrix represents the preferential members of every animal, it is easy to recognize animals which have strong interactions in the group. We use these approaches in demonstrating detection of sick and estrus (sexually active) cows in dairy herds as a use case of our approach. The variation

of the normal probability density function (PDF) of AIm at individual animal level as well as the Gaussian mixture model (GMM) of AIm at group level were explored to identify sick and estrus animals from the normal animals in the herd.

The remainder of this paper is organized as follows. While Section II discusses related works, Section III explains the theoretical steps of deriving AIm. In Section IV, the approach described in Section III is applied to a real dairy cattle mobility data set. Section V demonstrates the applicability AIm in detecting sick and estrus cows from a herd based on the simulated mobility data, including some directions to continue this paper further, and Section VI concludes this paper.

II. Related Works

The use of SNA has gained considerable attention in a wider range of fields such as sociology, business, and ecology [12], [19]. Exploring social interrelationships, quantifying disease transmission, and building models to explain dynamics in network topology is some of the highlighted applications. The study [12] emphasized that graph-based SNA is a promising tool for exploring such applications. Identifying and forming a network, developing methods for characterizing social behaviors, and exploring dynamic variability in social interrelationships are some of the key areas that animal science has been benefitted from SNA. Also, SNA has increasingly been used for improving the efficiency of human-managed animal farms.

Forming the network is one of the most important and challenging tasks in SNA. Because not only are social interactions heterogeneous but also are the factors which influence making interactions. Therefore, sampling animals in a network and defining their interactions must be done carefully as they contribute to improving the representativeness of the animal group and so the reliability of the derived social behaviors. One of the simplest ways of measuring social interaction is the use of NN identity. Rands [20] used NN identity data for assessing interactions based on a clustering technique in which a local group cluster matrix was developed in order to identify the most interactive nodes in a network. Evaluating the strength of the interactions among nodes is vital for conducting a comprehensive SNA. While the study [21] proposed weighted degree and strength centrality measures, taking into account the weight (strength) distribution of interactions, Cavanga *et al.* [11] discussed different mathematical approaches which can be used in computing the strength of the interactions in constructing the network.

The greater opportunities for monitoring social behaviors in a real-time manner have necessitated the development of novel SNA approaches through reexamining already existing techniques. As a result, various attempts have been made in deriving novel metrics in different applications. For instance, Qi *et al.* [15] developed a novel measure called Laplacian centrality to compute node importance and proved that it has greater performances than the standard centrality measure based on terrorist network analysis. Following the work in [15], the study [16] proved that using network-level information in quantifying node-level attributes can significantly improve the accuracy of selecting the top-k-most important nodes compared to the existing measures. These studies were, however, based on the static networks, but measures, which enable capturing time variant features, were highly demanded in SNA. Therefore, most research on SNA has focused on the exploring dynamic properties of social networks. For instance, while the study [18] derived a novel metric, dynamic centrality by exploring the limitations accounted in static network graph-based SNA, the time-ordered-graph method explained in [22] converts a dynamic network into a static network enabling the application of static SNA measures.

With the advancement of precision agriculture, applications of SNA for human-managed animals have drawn a considerable attention. For instance, significant improvements in farm management such as individual animal fitness, controlling decease transmission, and welfare could be achieved by exploring the social behavior of animals over a long period of time emphasizing automatic location measurements, NNs, and NN distance in SNA with farm animals [23]– [25]. The studies [1] and [2] emphasized the importance of using SNA, in particular, to dairy cattle management by exploring different characteristics such as community structure, social differentiation, stress, and productivity. However, Boyland *et al.* [2] highlighted that SNA concepts have widely been used for characterizing the different social behaviors of wild animals but not much for human-managed animals. Therefore, there are opportunities that SNA can be used in intensifying farm operations though, and they are not yet fully realized by the wider research community.

Our hypothesis is that SNA can effectively be capitalized in understanding and quantifying various factors such as husbandry practices, health issues, feeding, stress, and survival, thus making a significant impact on the stability of farm production systems and decision support tools. Moreover, the scale in which SNA can be applied is continuously increasing within the modern data monitoring tools. Hence, the necessity of deriving novel SNA approaches for applying to large-scale applications is also emphasized.

III. Theoretical Background

This section explains the process of deriving the entropy-based AIm. Initially, the procedure of computing the NN frequency matrix is discussed. This is followed by an experiment of the topological distance-based animal interaction range and we then discuss how to use network entropy to validate the interaction range. These two measures are then used in the process of computing animal importance to the group (i.e., AIm). Figs. 1 and 2 together give a graphical overview of the process in four steps. Then, in order to represent the variation of importance of individual and group levels in a distributional sense, the PDF and GMM of AIm are discussed. Finally, to test the validity of the derived measures, the system used to collect data and how the experiment was conducted, is explained.

A. Nearest-Neighbor Graphs

The spatial variability of NNs around each animal is commonly used to identify the most interactive individuals [20].

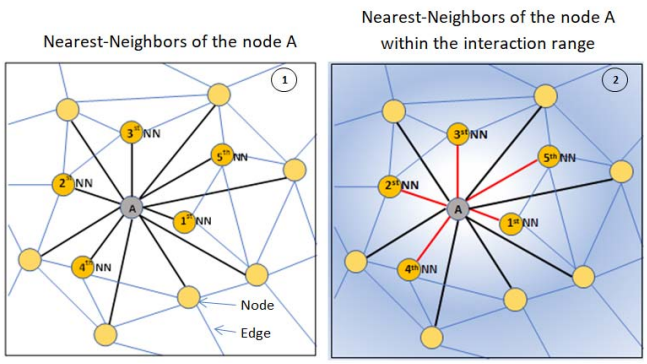


Fig. 1. NNs and interaction range. (a) NNs of the node A. (b) NNs of the node A with the interaction range.

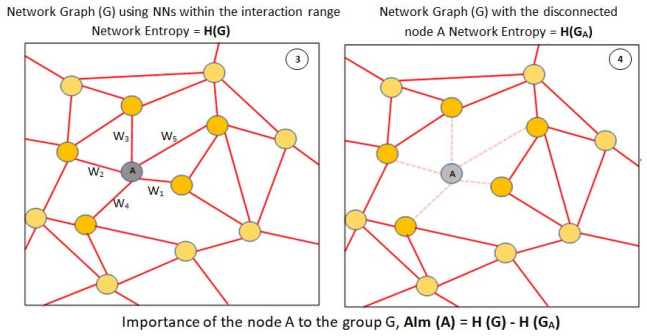


Fig. 2. Illustration of the technique of computing animal importance to a group, AIm. (a) Network graph (G) using NNs within the interaction range Network Entropy =**H(G)**. (b) Network graph (G) with the disconnected node A Network Entropy =**H($\mathbf{G_A}$)**.

Therefore, the NN approach was used as the foundation to understand the evolving cohesion (or social affiliation) with respect to each animal's daily activities and its interactions with other animals in the group.

We consider the mobility of a group of N animals over a period of time T. Then, we quantify the variation of animal relationships in every time window $t(\leq T)$: $T_w = T/t$ number of time windows. Information on NNs for each animal in this paper was determined based on the geographical distance between every pair of individuals in every t-min time window [Fig. 1(a)]. The frequency of each animal being the kth NN ($0 < k < N, k$ is the order of NN) to every other animal in the herd within the entire time period was formulated as a frequency matrix ($A_{N\times N}$). The kth-NN algorithm was used to identify the NNs of each animal in the group based on the interanimal geographical distances. The "*haversine*" formula [26] was used to compute the shortest distance between each of the locations of two animals. The matrix A was, therefore, defined as: $A = [f_{i,j}]$, where $f_{i,j}$ means the frequency of the jth animal being an NN to the ith animal over the period T. The matrix A, however, is not a symmetrical matrix, since interactions are not always symmetric. For instance, suppose the two animals i and j, and j is the first NN of i; then, the first NN of j would not be i whenever there is another animal closer to j than i. Hence, the interaction between i and j would not be symmetrical. The ith row of A represents the spatial variation in frequencies of the other $N-1$ animals being an NN to i.

In practice, it cannot be considered that each animal interacts with all other animals in the group in a particular time window because the strength of interactions would be very weak with distant animals (i.e., higher order NNs). Therefore, on average, the optimal number of interactions that every animal can have (i.e., the number of strong interactions) is essential in studying the intensity of interaction frequency of each animal within the herd and also constructing social network graph [Fig. 1(b)]. The interaction range metric derived below explains how to select those interactions.

B. Animal Interaction Range (k_r)

Sampling the most significant interactions in a network graph is one of the most crucial steps in graph-based SNA because the quality of the network graphs affects the robustness of social characteristics derived from them. The social influence range is the region where an animal exhibits strong interactions, and it depends on the individual's sociality within the group. Since sociality depends on various factors such as gender, age, and health, the social influence range can vary from animal to animal [25]. Therefore, an optimal measure to quantify social influence range is essential. Therefore, the main advantage of having an accurate social influence range is that it minimizes the loss of animal connectivity information while selecting interactions into a network graph. In this paper, the social influence range was defined as an animal interaction range (denoted as k_r) and represents the optimal number of interactions with high weights. The interaction weight was the reciprocal of the distance between a pair of interacting animals.

Although many studies such as [3] and [27] commonly used the metric distance for deriving the interaction range, Ballerini *et al.* [10] proved that topological distance, i.e., interaction range with number of animals, is more robust than the metric distance. Therefore, this paper used the topological distance approach to derive the interaction range. The advantages of using this approach are the density fluctuations of animal aggregation, which can be well explained compared to metric distance, and the metric distance, which can be derived from the topological distance. Therefore, the interaction range was quantified based on the animal anisotropy factor and the network entropy. The network entropy-based k_r was used to confirm the result obtained from the anisotropy factor.

1) Anisotropy Factor (γ)-Based k_r Estimation: Based on the spatial distribution of NNs and their orientation around each animal, the anisotropic structure of a moving group of animals varies with increasing order of NNs (i.e., k). The anisotropy value (γ) represents the effect of interaction among animals, whatever the interaction is, and quantifies to what extent the spatial variation of the kth NN (around a reference animal) is anisotropic. This is subsequently used to determine k_r, regardless of the distance between the animals.

Initially, animal locations in this paper were converted into 3-D Cartesian coordinates (X, Y, Z) and a set of NN vectors (u), which were derived as

$$u^{(i,k)} \in u, u^{(i,k)} = \left[\left(u_x^i, u_y^i, u_z^i\right)\right]_{(1\times 3)}$$

where $u_{i,k}$ is a unit vector directed toward the kth-order NN of the animal i (here $i = 1, 2, \ldots, N$). Also, the animal group center velocity vector (V) was also calculated as

$$V = [v_x, v_y, v_z]_{(1\times 3)}$$
$$v_x = \frac{1}{N}\sum_{i=1}^{N}\frac{x_i}{t}, \quad v_y = \frac{1}{N}\sum_{i=1}^{N}\frac{y_i}{t}, \quad v_z = \frac{1}{N}\sum_{i=1}^{N}\frac{z_i}{t}$$

where t is the width of the time window. Then, normalized u and V (u_n and V_n) were computed as

$$u_n^{(i,k)} = \frac{u^{(i,k)}}{\sqrt{\left(u_x^i\right)^2 + \left(u_y^i\right)^2 + \left(u_z^i\right)^2}}, \quad V_n = \frac{V}{\sqrt{v_x^2 + v_y^2 + v_z^2}}.$$

Second, in order to calculate the orientation of the kth-order NN vector of the ith animal (i.e., $u_{i,k}$) with respect to V_n, the projection matrix $M_{i,k}$ of the i_{th} animal was calculated as

$$M_i^k = \left[u_n^{(i,k)} u_a^i\right]_{(3\times 3)}, \quad \text{for} \quad i = 1, 2, 3, \ldots, N$$

where $u_a^i \in u_n^{(i,k)}$ for $a = \{x, y, z\}$. Then, $M_{i,k}$ of all animals was averaged ($M^k = (1/N)\sum_{i=1}^{N} M_i^k$) to derive a matrix, which projects along the average direction of the kth-order NNs of all animals in the herd. M_k corresponds to the most relevant directions in the space and its eigenvalues represent the number of NN vectors that can be detected along the direction of the corresponding eigenvectors [11]. The anisotropy factor for the kth-order NNs, $\gamma(k)$ was calculated at the third step by taking the square of the scalar dot-product between the normalized eigenvector (e_k), which corresponds to the minimum eigenvalue of M_k and the normalized herd center velocity (V_n) as

$$\gamma(k) = (e^k . V_n)^2. \tag{1}$$

The process was repeated varying the NN order, i.e., $k = 1, 2, \cdots, N$, in every time window and averaged over T_ws to obtain the variation in anisotropy γ with respect to the spatial distribution of NNs (i.e., with increasing order of NN). The eigenvector corresponding to the smallest eigenvalue was used here since there is a reduced probability to detect an animal's NNs along the direction of this eigenvector. Finally, according to [10], for the isotropic case, the value of γ is 1/3 (i.e., no interaction) and the k value at which this occurs is defined as the value of interaction range (i.e., the value of k_r). Thus, interactions between the reference animal with its NNs were deemed not significant after the k_rth-order NNs (i.e., isotropically distributed). In other words, the topological distance threshold is the k_r value, which is the number of individuals deemed to be around an animal. The metric distance is the distance to the k_rth NN from the focal animal. More details on the anisotropy factor and its derivation are given in [10] and [11].

2) Entropy-Based k_r Estimation: Variability in the daily group entropy, which is the average entropy (explained in Section III-C) over all the time windows in this paper, was computed as follows. Network graphs were generated by increasing the NN order (i.e., $k = 1, 2, \cdots, N-1$) around each animal. The increasing rate of herd entropy was examined over the time period T, and the k_r value was selected in units of animals up to where there was no significant incremental increase in herd entropy.

C. Network Graph-Based Animal Importance to the Group

Network graphs illustrate the structural connectivity of social groups and are more complicated due to the heterogeneous nature of social interactions [Fig. 2(a)]. Therefore, including only the most relevant interactions, for instance, interactions belong to the interaction range (i.e., within k_r), would help to simplify the network complexity and would also enhance the quality of the structural information about animal connectivity acquired for characterizing different social behaviors, such as density, centrality, and communicability [12]. Such graph-theoretic measures can be applied directly to quantify these behaviors. Here, we follow an approach combining the network graph entropy and weighted degree centrality to evaluate animal importance to the group (denoted as AIm). For instance, Fig. 2(a) represents the network graph (G) built by selecting the interactions, which belong to the interaction range of each node.

The variability in the number of interactions made by an animal depicts its influence to the group, because the removal of a highly interactive animal causes a substantial change in the structural group properties (e.g., group cohesion and connectivity). The animal importance to the group, AIm, quantifies to what extent an animal influences the group. This can be measured as the group entropy variation caused by the removal of that animal from the group. The entropy (H) characterizes the structural information based on a group level as well as on an individual animal level [16]. From a mathematical point of view, the entropy of a random variable X, $H(X)$ is the expected information encoded within X [28] and formulates as

$$H(X) = E[I(X)]$$

where E is the expectation and $I(X)$ denotes the information contained in X and is computed as $I(X) = (1/P(X))$, where $P(X)$ is the probability of X. Thus, according to the definition of statistical expectation, $H(X)$ can be simplified into

$$H(X) = -E[P(X)] = -P(X)\log P(X).$$

Following the entropy of a single random variable, the entropy of a system which consists of a set of n random variables can be computed as

$$H(X) = -\sum_{i=1}^{n} E[P(x_i)\log(P(x_i))]$$

and this quantifies the expected information carried by the whole system.

To compute the entropy of the group [i.e., $H(G)$], a network graph was constructed considering only the interactions among individuals within the interactive range. Mobility of animals in this paper was considered as random variables and the amount of structural information attributable to each animal represents its influence to the herd. The direct weighted degree centrality (C_W), which is a local (i.e., individual or node level) measure used in graph theory, was used to quantify an individual animal's influence on the herd. This measure reflects to what extent an animal is directly connected with other animals based on the weight of direct interactions [21]. There are two types of direct interactions associated with an animal: the interactions it initiates (i.e., outward) and the interactions it receives from others (i.e., inward). In this paper, the number of direct interactions was the sum of the both inward and outward interactions. Assuming an arbitrary time window t_i, suppose the ith animal has υ number of direct interactions with weights $\{w_j\}_{j=1}^{\upsilon}$. The weight of an interaction was computed as the reciprocal of the distance between the pair of interacting animals. Then, weighted centrality (C_W) of the ith animal, $C_W(i)$ was computed as $C_W(i) = \sum_{j=1}^{\upsilon} w_j$, for $i = 1, \cdots, N$, and then normalized (C'_W) to compute the group entropy, $H(G)$ as follows. C'_W satisfies all the conditions to be a probability distribution

$$H(G) = \log\left(\sum_{i=1}^{N} C_W(i)\right) - \sum_{i=1}^{N} C'_W(i)\log(C_W(i))$$

where

$$C'_W(i) = \frac{C_W(i)}{\sum_{i=1}^{N} C_w(i)}.$$

The importance of the ith animal to the group, $AIm(i)$, was computed as the change in group entropy caused by the removal of the ith animal from the group, that is,

$$AIm(i) = H(G) - H(G_i) \tag{2}$$

where G_i represents the group without the ith animal. Also, Fig. 2 graphically illustrates the computation of AIm for the node A. The importance value of each of the N animals was computed for all time-windows and then averaged over the time period T to obtain the importance metric for that animal.

To compare the reliability of entropy-based AIm with the other existing measures, the AIm value of all animals was computed using C_W, which uses only the local-level information and the Laplacian weighted centrality (C_L) proposed in [29]. C_L of a node in a weighted network graph (interaction strengths are given as weights) defined as the relative drop of Laplacian energy due to deactivation of all interactions of that animal with others in the group. The study [29] derived an approach to compute it by using intermediate-level connectivity information, i.e., the interactions of an animal with first and second NNs. The C_L formula is explained briefly below and we refer to [29] for the derivation and more details about C_L

$$C_L(i) = 4C_W^{c(i)} + 2C_W^{m(i)} + 2C_W^{h(i)}, \quad for \quad i = 1, 2, \cdots, N \tag{3}$$

where $S(i)$ contains the set of first NNs of the ith animal and $W_{i,j}$ is the weight of the interaction between i and j; $C_W^{c(i)} = \sum_{j\in S(i)} W_{i,j}^2$ is known as the number of closed two-step interactions containing the animal i (i.e., from i to j and j to i); $C_W^{h(i)} = \sum_{j\in S(i)}(\sum_{l\in\{S(j)-i\}} W_{i,j}W_{j,l})$ is the two-step interactions containing the animal i as an edge vertex of the path (i.e., i to j and j to l, where ls are the first NNs of entries of the set $S(i)$; and $C_W^{m(i)} = \sum_{j,l\in S(i),\&j\neq l} W_{j,i}W_{i,l}$, represents the two-step interactions containing the animal i in the middle of the path (i.e., j to i and then i to l).

D. Gaussian Mixture Models

In order to get an overall idea about the variation in animal importance of a group, the joint distribution of AIm of all members was derived using the GMM approach. This approach was based on the assumption that the AIm of each animal has a Gaussian probability distribution (hereafter, termed PDF). In this paper, the PDF of the AIm of an animal represents the distribution of AIm values collected over a period of time. GMM is a parametric PDF of a set of PDFs and is computed as their weighted sum. This can also be considered as a hybrid version of a set of PDFs and provides not only a smooth overall distribution but also its components unveil the details about the multimodel nature of the density.

Suppose the mean and covariance of a D-dimensional continuous Gaussian random variable X is $\mu_{(1\times D)}$ and Σ, respectively. Theoretically, the PDF of X is denoted as $X \sim \mathcal{N}(\mu, \Sigma)$ and written as

$$f(X|\mu, \Sigma) = \frac{1}{(2\pi)^{\frac{D}{2}}|\Sigma|^{\frac{1}{2}}} e^{\left(\frac{1}{2}(X-\mu)^T\Sigma_i^{-1}(X-\mu)\right)}.$$

Then, the GMM of a set of m such random variables is written as

$$F(X|\Phi) = \sum_{i=1}^{m} w_i f(X|\mu_i, \Sigma_i)$$

where the GMM model weights, w_is satisfy $\sum_{i=1}^{m} w_i = 1$ and $\Phi = \{w_i, \mu_i, \sigma_i\}$, $i = 1, \cdots, m$, are the set of parameters in GMM model that has to be estimated.

To estimate Φ, the iterative expectation–maximization (EM) and maximum A-posterior estimation techniques are commonly used. Depending on the different characteristics of the set of PDFs, GMM can have various forms. The covariance, Σ, could be similar for all components in some cases, whereas it is possible to use a full rank or diagonal covariance matrix. In our evaluation, a full covariance matrix method is used and the EM technique is used to estimate the parameters. Moreover, the GMM model configurations depend on the available data as well as the application. We refer to [30] for more details about GMM.

For instance, if $X_1 \sim \mathcal{N}(\mu_1, \Sigma_1)$, $X_2 \sim \mathcal{N}(\mu_2, \Sigma_2)$, and $X_3 \sim \mathcal{N}(\mu_3, \Sigma_3)$ with $D = 1$, then the GMM of X_1, X_2, and X_3 (say X_4) can be written as

$$F(X_4|\Phi_4) = w_1\mathcal{N}(\mu_1, \Sigma_1) + w_2\mathcal{N}(\mu_2, \Sigma_2) + w_3\mathcal{N}(\mu_3, \Sigma_3)$$

where $\Phi_4 = \{(w_1, w_2, w_3), \mu_4, \Sigma_4\}$ and $w_1 + w_2 + w_3 = 1$.

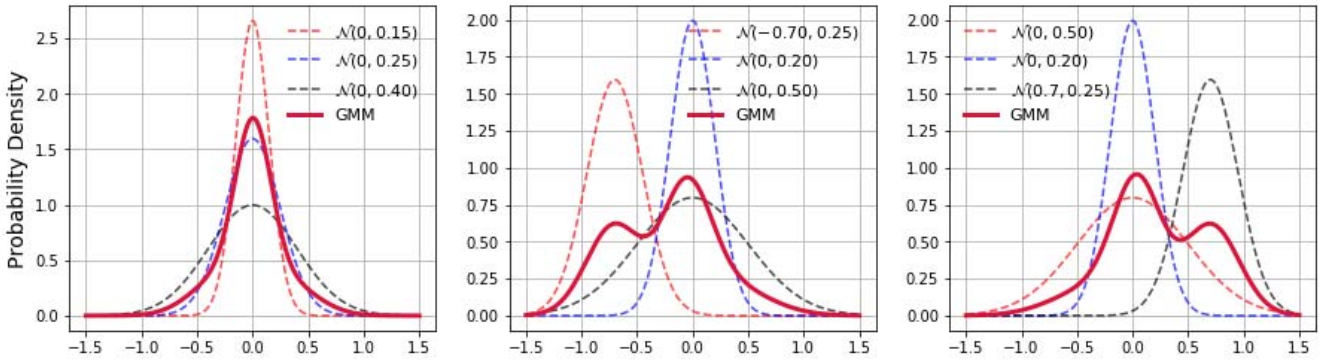


Fig. 3. Variation of the shape of GMM in response to the significant change of the mean of PDFs. First one for similar μs and different σs. The second and third graphs represent the change of the shape of GMM when one PDF is shifted to the left and right of the other PDFs two, which have similar μ and different σs, respectively.

Fig. 3 represents the variation of the shape of the GMM of three different Gaussian random variables X_1, X_2, and X_3 (where we assume $D = 1$ and $\Sigma_i = \sigma_i$ for $i = 1, 2, 3$) in response to the change in their PDFs for three different cases: 1) similar means with different variances ($\mu_1 = \mu_2 = \mu_3$) (left); 2) variance of X_1, X_2, and X_3 are different, and X_2 and X_3 have similar means, but X_1s mean diverged to the left of the mean of X_2 and X_3 ($\mu_1 < \mu_2 = \mu_3$) (middle); and 3) the opposite case of the second case, but the change happens with the variable X_3 instead of X_1 ($\mu_1 = \mu_2 < \mu_3$) (right). In each case, the variance of X_1, X_2 and X_3 are different from each other, i.e., $\sigma_1 \neq \sigma_2 \neq \sigma_3$. Therefore, the GMM represents a multimodel nature (in this case it is a bimodal because there are two-packs only) mainly when the means of PDFs are significantly different.

When considering the AIm of an arbitrary animal as a random variable similar to X explained earlier with $D = 1$, the PDF of the AIm of that animal can be written as $f(X|\mu, \sigma) = (1/2\pi\sigma^{2(1/2)})e^{(((X-\mu)/2\sigma))^2}$, i.e., $X \sim \mathcal{N}(\mu, \sigma)$. Then, the GMM of AIm represents the joint distribution of the AIm of the group.

E. Geolocation Mobility Data

GPS location data of 33 (N) cows (a herd) were collected over a 24-h period for five consecutive days (T) at a Teagasc research dairy farm, in Moorepark, Ireland. All cows were part of a study on robotic milking in grazing production systems [31]. The data were partitioned into nonoverlapping time windows each of 10 min (t) duration (i.e., $T_w = 720$ windows for the entire duration). For each animal, maximum of three locations were observed in every 10 min, so that the average location of each cow was computed within each window. Linear interpolation was used to compute missing observations. The cows grazed different paddocks during the 5-day period with varying distances from the milking parlor (Fig. 4). All cows visited the milking station at most twice daily and moved to a new paddock after each milking.

IV. Results

The applicability of the mathematical procedures is discussed by using a GPS mobility data set of a dairy herd

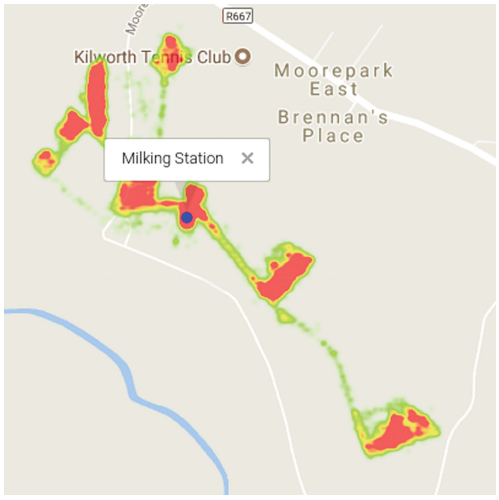
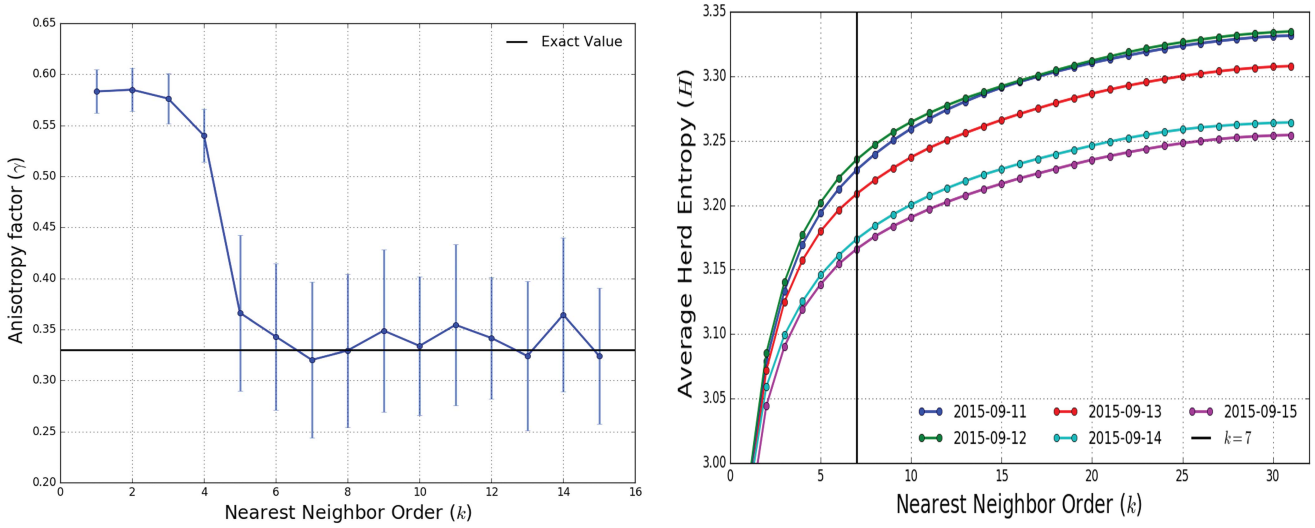


Fig. 4. Heat map of the GPS locations of the cows at the research farm in 5-day period. Blue dot: milking station.

explained in Section III. First, the cattle social interaction range is derived using the data set and it is then validated based on the variability of network entropy. The variability of social affiliation of dairy cattle is discussed next. Finally, the entropy-based AIm is computed and its validity is compared with the importance evaluated using the direct degree and Laplacian centrality measures. MySQL database was used for efficient data storage and access, and Python software was used for data analysis and the simulation study. More specifically, the Python packages *networkx* and *scikit-learn* were used for building the network graph and computing GMM models, respectively.

A. Interaction Range of the Pasture-Based Dairy Animals

The anisotropy factor (γ) revealed a clear relationship with the herd center velocity (V) along the direction of the eigenvector. This corresponded to the smallest eigenvalue of the average projection matrix (M') [Fig. 5(a)]. The γ decayed with increasing NN order (k) and reached 1/3 (exact value or isotropic point) near a k value of 7. Thus, the NNs above the seventh-order NN were isotropically distributed around a focal animal and did not strongly interact with it. Therefore,



(a) (b)

Fig. 5. Derivation of dairy cattle social interaction range (k_r). (a) Anisotropy variability with NN order (k) for the 33 dairy cows. (b) Herd entropy variation with the number of NNs for each day [black vertical line: interaction range derived from anisotropy factor (γ)].

the value of interaction range (k_r) was selected as 7 ($k_r = 7$). In other words, the strength of the interactions of a focal cow with the first seven NNs was considerably higher compared to the interactions beyond the seventh NN.

The variation in average herd entropy with the spatial distribution of NNs over the 5 days of the study is shown in Fig. 5(b). Although the herd entropy increased with the NN order (i.e., k), it did so at a declining rate. This is because entropy was calculated based on the weighted degree centrality, C_W. The weights of interactions reduced with the increasing order of NN. Therefore, the contribution to the increment of herd entropy, $H(G)$ from the interactions with the animals beyond the k_rth NN was minimal (on average ≤ 0.01). This indicates that the interaction range of our pasture-based dairy cows is 7.

To be more precise about the interaction range derived from Fig. 5(a), the effect of the interaction range on the herd entropy was examined. The herd entropy characterizes the information about the connectivity of the herd. Hence, the broader interaction range will increase the herd connectivity and so will the herd entropy. However, the interaction strengths weakened as the interaction range increased and, consequently, contributed to a decline in the increasing rate of herd entropy. Therefore, the optimal interaction range (i.e., number of NNs) which does not make any significant contribution to improve the herd entropy should be the optimal interaction range [Fig. 5(b)]. Although having a broader interaction range brings some disadvantages such as higher competition for food and space and risk of falling sick [32], there are some benefits as well, especially in graph-based SNA and also for corporate defending against predator attacks.

B. Time-Evolving Interactions With NN Frequency Matrix

The frequency of each cow being within the interaction range of every other cow over the 5-day period of the study is represented in Fig. 6, which can be read similar to reading a normal square matrix. For instance, considering the cow index (ID) 10, the corresponding row represents the frequencies of interactivity of all other cow IDs with the cow ID 10, while the column depicts the frequencies of interactivity of cow ID 10 with the remaining cow IDs, including itself. Since the NN frequency matrix is not a symmetrical matrix, the entries of a particular row were not always exactly similar to the entries of the corresponding column. As an example, the interactivity of the cow ID 10 to the cow ID 9 was greater than 70%, whereas it was less than 60% for the cow ID 9 being interactive with the cow ID 10 over the 5-day period of the study.

The variability of frequency fluctuations provided a clear illustration of the intensity of interanimal interactions (social affiliation) in order to identify specifically the most and least interactive cows. In general, most of the frequency values were below 40% (i.e., less than 30 out of 120 h), but few cow ID pairs represented greater than 50% values in Fig. 6. For instance, the interactivities of the cow ID pairs 7–8, 13–15, and 29–31 were between 60% and 70%, and it was greater than 80% for the cow ID pairs 22–23, 23–24, and 24–25.

Since the highly interactive cows are easily recognizable in Fig. 6, this matrix could be used to select especially the most (or least) cohesive cows in a herd. Hence, this matrix could be useful in characterizing herd-level behaviors such as the cows which are at high (or low) risk of getting viral infections, the leading disease carriers, sexually active cows, and the cows which are highly likely to synchronize their activities. Such information would be useful to manage herds more efficiently in different farm operations such as maintaining optimum animal well-being and training cows for voluntary participation in milking. In SNA point of view, sampling animals form a larger group to form network graphs.

C. Animal Importance to the Herd

The importance (AIm) of the 33 study cows from the weighted degree centrality and Laplacian centrality measures

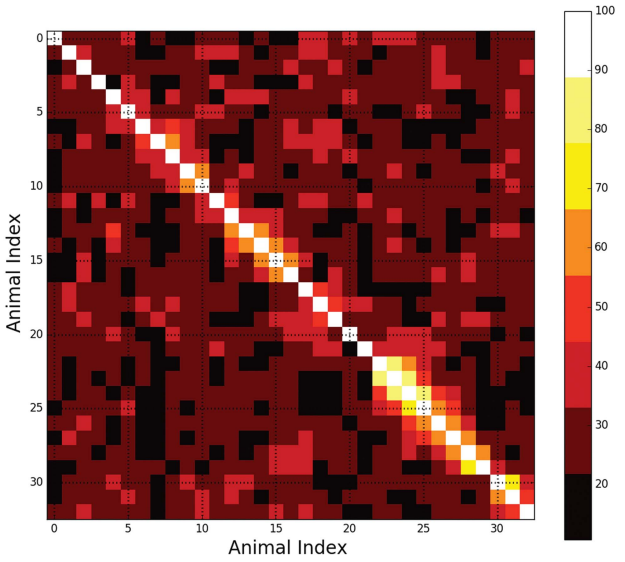


Fig. 6. Frequency of being individual's interaction range as a percentage of the total number of time-windows (i.e., 720) over the 5-day period [the size of a time-window was 10 min and the frequency percentages were computed as $(f/720) \times 100$].

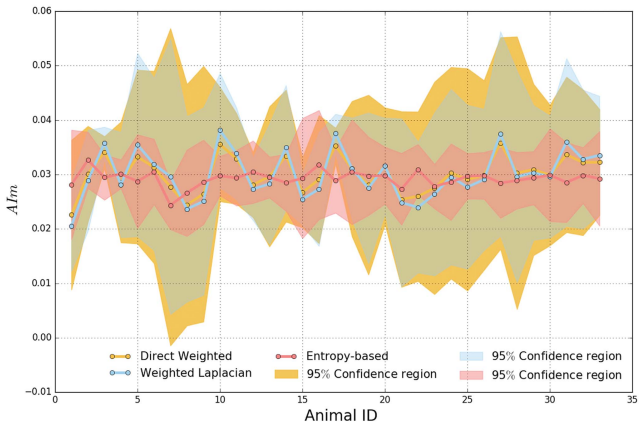


Fig. 7. AIms of 33 study cows and their variability from the direct weighted centrality (C_W), weighted Laplacian centrality (C_L), and herd entropy-based approach.

(C_W and C_L) and herd entropy variation over the 5-day study period are shown in Fig. 7. The AIm based on the C_W and C_L measures were more or less similar and fluctuated more with greater variability compared to the herd entropy-based AIm. The C_L-based AIms were more stable than the AIm computed using C_W because the confidence limits of the C_L-based AIm were slightly tighter than the confidence limits of the AIm computed using C_W. On average, the confidence limits of the herd entropy-based AIm was considerably tighter compared to C_W and C_L measures based AIms. Therefore, the stability of the AIm computed using the entropy-based measure is greater than the other two measures.

The degree centrality is itself a measure of animal importance to the herd and can be computed mainly using the direct and weighted degree centrality (C_W) measures. C_W takes into account the heterogeneous nature of social connectivity based on their interaction strengths, but the direct degree centrality assumes that all interactions have the same strength. Hence, C_W is a commonly used reliable measure for quantifying animal importance in SNA. Moreover, the accuracy of C_W-based AIm could even be improved by deriving the interaction strengths (weights) from well-defined methods [21] rather than interanimal distance. Although C_W performs well in quantifying animal importance, based on local-level connectivity information, it has been reported that the reliability of animal importance measure could be improved further by increasing the range of connectivity information based on the C_L measure defined by [29].

However, C_L uses only the intermediate-level information (i.e., the connectivity data of each animal associated with the single (direct) and two-step interactions). Therefore, to improve the robustness of AIm, this paper considered the global-level information (i.e., herd entropy) to compute animal importance because highly reliable AIm is necessary for detecting animals which have significant influence, such as dominant cows and group leaders. That is why the uncertainty of the entropy-based AIm of the 33 study cows was considerably smaller (with a narrower confidence interval) compared to C_W and C_L in Fig. 7. Thus, this guarantees that the entropy-based AIm characterizes the individual cow influence to the herd more precisely than the existing measures.

V. Discussion

The "datafication" of modern-day dairy and other agricultural production systems, through the widespread adoption of sensors [33], facilitates the development of novel metrics and algorithms to detect individual animals that may require particular attention. This is, especially true with the proliferation of these technologies which, because of their ever-reducing costs are now being used on an individual animal basis. It was our hypothesis that a greater exploitation of the data from all individuals in a group could provide more information than the analysis of the data relating to just a single animal in the process of evaluating AIm. In this section, we discuss two use cases of AIm and directions for further studies based on the limitations and benefits of the AIm approach.

A. Animal Mobility Simulation to Identify Estrus and Sick (or Injured) Animals

In order to demonstrate how to use AIm measure in a real-world application, an example is illustrated regarding the detection of sick and estrus cows in a herd. The illustration is, however, based on the simulated mobility data (using the *pymobility* Python package [34]) because the data set used earlier did not have any estrus or sick cows and was too small and was collected only over a short period of time.

In the simulation study, the behavior of sick and estrus cows was compared to normal contemporaries in a herd based on their variability in the AIm metric. Two steps were taken in carrying out the simulation study. First, the mobility patterns of the sick and estrus cows relative to the normal cows (i.e., nonestrus and healthy) were simulated separately

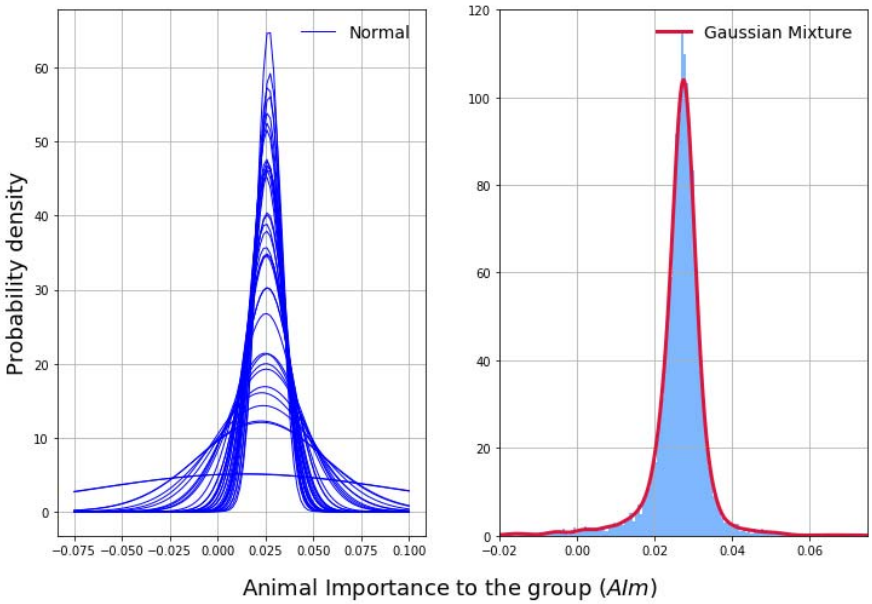


Fig. 8. Individual PDF of AIm of normal herd animals(left) and their joint PDF (right).

over a time period by using a random-way-point mobility pattern [34]. This approach is commonly used to simulate animal movements within a given region, depending on their velocity and waiting time (time spent at a position before making the next random movement). Therefore, variability in the characteristics of walking area, speed, and waiting time was considered as the signs (parameters) in simulating mobility of sick (and estrus) cows. Second, to represent the variability in AIm, the PDF of AIm was computed for each sick animal, each animal in estrus, and each normal animal. Also, to illustrate the group-level variation of AIm of sick and estrus herds with respect to normal herds, the joint PDF of the individual PDF was derived.

Three herds (i.e., normal, sick, and estrus animals) each with 40 cows were simulated separately within a 60m $\times$ 30m (rectangular paddocks are preferred with 2:1 width and depth ratio)[1] paddock over 1500 time windows ($\sim$10 days). In sick and estrus herds, 35 of the cows were assumed to be normal and the rest were assumed to be either sick or in estrus. In this simulation, we did not consider sick and estrus animals together in the same herd. This is because there might be a cow in both conditions and it might be difficult to differentiate between such conditions in the same animal.

According to the literature, the average cow speed is 2–3 km/h and it can be increased up to 4.5 km/h in properly maintained farms. However, these ranges are affected by many factors such as paddock and track design, weather conditions, and distractions. [1],[2] In optimal conditions, cows lie down generally for 14 h per day ($\approx$ 30min/h) and on average, a normal cow gets up 16 times a day. The resting time varies for many reasons such as age, heat cycle, health, weather, herd size, and the housing and paddock conditions.[3] Therefore, in the simulation study, cow velocity and the waiting time at a position were assumed to vary uniformly in the ranges of 0.0–3 km/h and 0–30 min/h, respectively. Also, we assumed that normal cows move over the entire paddock at their own peace.

As sick cows move slowly and take more time to make the next move, they have long resting (waiting and lying) time. Therefore, to simulate the mobility pattern of a sick cow, the region covered and the velocity range were reduced by half and the waiting time was doubled compared to a normal animal. This procedure was repeated for all time windows and the PDFs of AIm for each animal were quantified. The same process was followed for the herd with cows in estrus by changing the mobility pattern of cows in estrus; since the level of activity intensifies when cows are in estrus [35], the velocity range was doubled and the waiting time was halved. These cows were allowed to move in the entire paddock the same as the normal cows in each iteration.

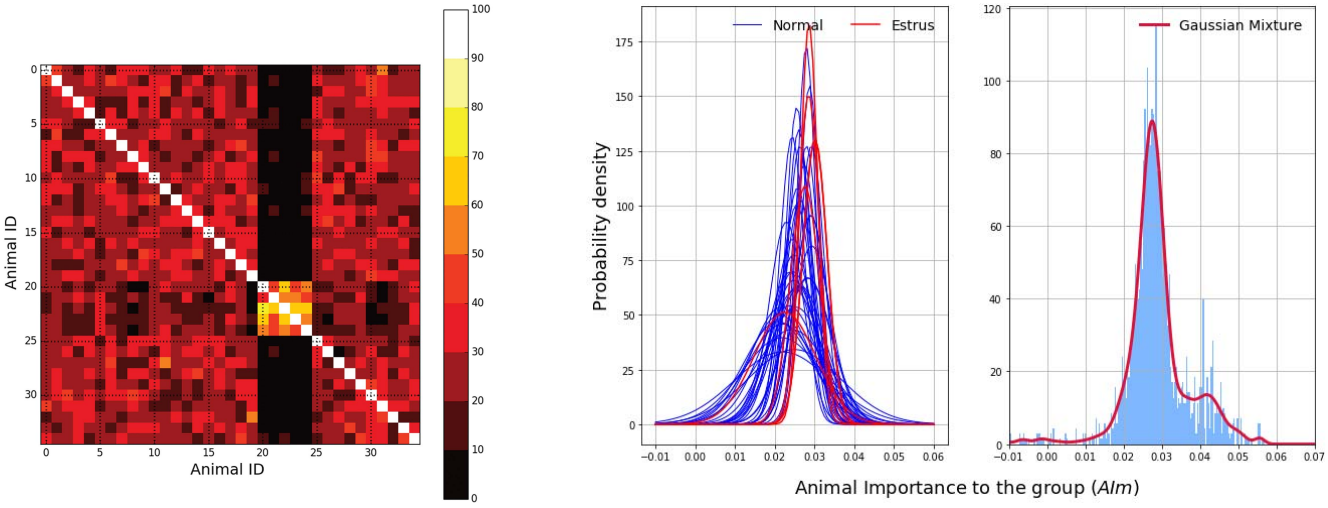
1) Normal Herd: Fig. 8 represents the PDF of AIm of individual animals (left) and their joint PDF (right). In general, all PDFs are distributed around a common mean (approximately 0.025) with different shapes (variances). The corresponding mixture (joint) PDF represents the overall distribution of AIm of the herd and does not indicate any multimodel nature as there is only one peak similar to the theoretical example explained (first case) in Fig. 3. Therefore, any deviation from this behavior would be an early indication of herd approaching an unusual behavior and thus needing attention. Sections V-A2 and V-A3 illustrate the nature of such deviation when there are sick and estrus animals in the herd.

2) Identifying Cows Approaching or in Estrus: Fig. 9(a) and (b) present the simulated outcomes of the variability in interactivity and AIm (as PDFs) of a mixture of estrus

[1]https://www.teagasc.ie/media/website/animals/dairy/GrazingInfrastructure.pdf

[2]https://www.dairynz.co.nz/media/214 237/Understanding-cow-movement.pdf

[3]http://www.milkproduction.com/Library/Scientific-articles/Housing/Cow-comfort-9/



(a) (b)

Fig. 9. Variation of social affiliation and importance of the estrus and normal cows. (a) Intensity of interactivity of cows in estrus (IDs 20–24) with the nonestrus animals. (b) Individual PDF of AIm of the estrus animals with their normal herd animal (left) and their joint PDF (right).

and nonestrus cows, respectively. In general, the intensity of interactivity between the nonestrus cows ranged between 30% and 50% [IDs 20–24 represent the estrus cows; Fig. 9(a)], and on average, the intensity of interactivity among only the cows in estrus was greater than 70% but was less than 10% between the estrus and nonestrus cows [columns of the estrus cows; Fig. 9(a)]. On the other hand, the intensity of interactivity of normal cows to the estrus cows (rows of the estrus cows) was less than 30% [Fig. 9(a)]. The mean values of the PDFs of nonestrus cows were approximately similar (i.e., the PDFs distributed around a similar mean value), but their variance was different [Fig. 9(b)].

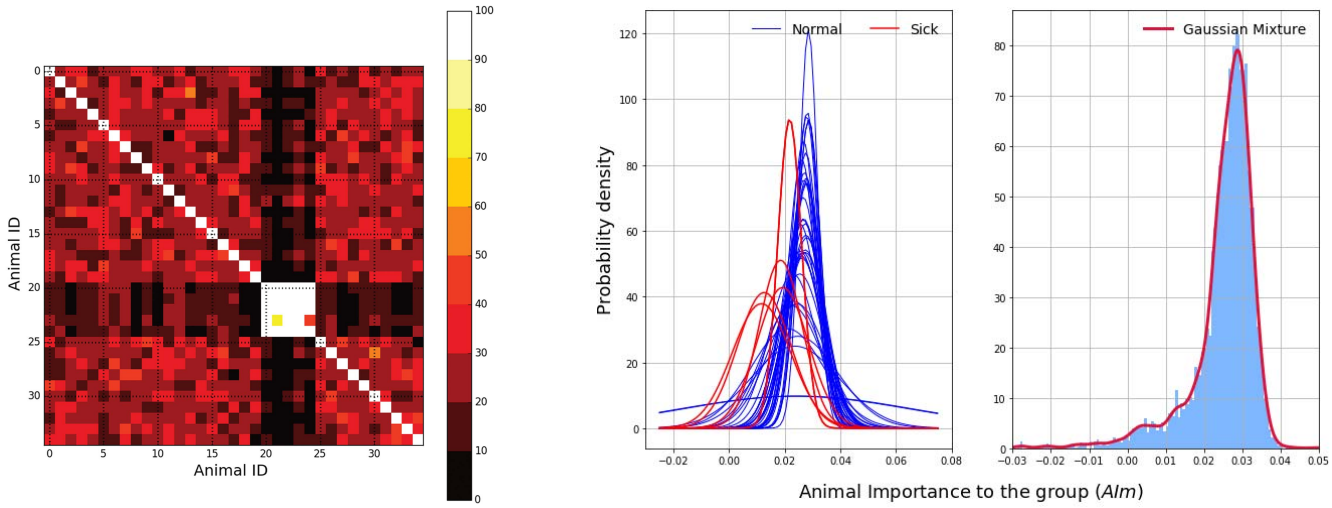
The PDFs of the estrus cows were shifted toward the right of the PDFs of the normal cows and denser (i.e., less variance and high kurtosis) than normal cows. Consequently, the joint of these PDFs represents a bimodal behavior as there are two peaks in the distribution. In comparison to the GMM of normal herd given in Fig. 8, the GMM is expanded toward the right with a peak value. This is due to the cows in estrus having higher AIm and validates the theoretical illustration (case 3) given in Fig. 3.

The mobility patterns of the sexually active cows are generally different from the normal cows due to their greater tendency to join groups of cows that are also in estrus, and thus have less resting time, as well as ending up walking together. Consequently, the measured interactivity among cows in estrus is greater than the interactivity between cows in estrus and nonestrus. Hence, the simulated intensity of interactivity between the cows in estrus was greater compared to the intensity of interactivity between estrus and normal cows in Fig. 9(a). Consequently, cows in estrus become the most important (i.e., highly cohesive) members of the herd more frequently than normal cows. As a result, less variance and higher mean of their AIm is detected. Therefore, the PDFs of AIm of cows in estrus become denser compared to that of the normal cows [Fig. 9(b)], and also, the joint PDF of AIms is more expanded toward the right compared to normal the herd (Fig. 8).

The failure of detecting cows in estrus in herds not adopting fixed-time artificial insemination has negative repercussions for farm profit due to extended calving intervals as well as possibly the cost of semen inseminated at the incorrect time of the estrus cycle [33]. Roelofs *et al.* [36] noted that herd behavior should be closely monitored as some of the well-documented signs of estrus may not exclusively be exhibited by the cows in estrus. Among the different herd behaviors which have been categorized as primary and secondary signs of estrus [37], observing the change of cow mobility patterns can be useful as a behavioral sign for generating early alerts about the cows approaching or in estrus. Therefore, the matrix A and AIm can be used in detecting early estrus signs.

3) Identifying Sick or Injured Cows: Fig. 10(a) depicts the variability of intensity of interactivity for a mixture of sick and normal cows. The variability in the intensity of interactivity of sick cows (20–24 rows and columns) is clearly highlighted in Fig. 10(a) evident. The intensity of interactivity of sick cows with the herd was below 40%, whereas, among the sick cows, it was greater than 80%. The PDFs of the sick animals shifted toward the left of the mean value of the PDFs of normal cows [Fig. 10(b)], and also, their mean values were different to the mean values of the PDFs of normal cows. As a result, the overall distribution of the AIm of the herd is expanded to the left side with a peak, i.e., bimodal behavior with left-tailed distribution.

Due to the fact that sick cows are reluctant, or have difficulty in moving, they might not always be able to follow the herd. They may become isolated from the herd or lag behind the herd. Consequently, their mobility patterns may be different (or diverge) from their peers in the herd. This divergence can, therefore, be easily used to differentiate those animals. In SNA, this divergence of mobility can be accounted for as a decline in their interactivity with the herd. Hence, highly interactive cows will generally not be the sick cows. Also, they will have fewer chances of being as highly interactive members to other herd mates. Therefore, the intensity of interactivity of sick cows with the herd generally becomes low [below



(a) (b)

Fig. 10. Variation of social affiliation and importance of the sick and healthy cows. (a) Intensity of interactivity of sick cows (IDs 20–24) cows with healthy cows. (b) Individual PDF of AIm of the sick animals with their normal herd animals (left) and their joint PDF (right).

30% in Fig. 10(a)]. Moreover, sick cows may gather as a subgroup, so that they are counted as being highly interactive with each other while computing the NN frequencies. That is why Fig. 10(a) represents the greater intensity of interactivity (greater than 80%) values for sick cows. However, in practice, this does not imply that they are exclusively associated with each other. As sick cows experience different mobility patterns and are less interactive within the herd over time, their AIm becomes less and varies more compared to normal cows because AIm depends on the number of interactions and their weight. Hence, the mean values of their PDFs of AIm were less and different from the healthy cows in Fig. 10(b). Consequently, the joint PDF of AIm is expanded more toward the left compared to the normal herd joint PDF given in Fig. 3. This validates the theoretical concept explained under the case 2 in Fig. 3.

Early detection of health-related ailments is extremely important to prevent erosion of herd profit but also to minimize the impact on animal welfare. In the early attempts of animal health assessment studies, animal mobility was commonly used to identify animal health issues [38]. Moreover, Rahaman *et al.* [39] stated that automatically recording mobility behaviors facilitates the derivation of value of animal health alerts. Apart from the cow mobility pattern analysis, lying and standing behaviors [40], body temperature and visual signs [40], and variation in hunger [41] have also been used to evaluate animal health status.

4) Herd Behavior With Increasing Sick and Estrus Animals: To explore the herd behavior as more animals become sick in the herd, under the same settings, individual AIm and herd entropy were computed by varying the number of sick animals as a percentage of the herd (0%–70% prevalence within the herd). To represent variability in AIm and herd entropy, the 95% confidence limits were also computed. It was assumed to remain that animals falling sick were sick for the rest of the simulation process. The same procedure was repeated with an increasing number of estrus animals in the herd. As cows are in estrus for only a few hours, we did not, however, assume that estrus animals stay to remain in estrus for the rest of the simulation.

The variability in average herd entropy with an increasing number of sick and estrus animals (as a percentage of herd size) in the simulated herd is shown in Fig. 11(a). The herd entropy of the normal herd (i.e., 0% sick or estrus cows) was 3.58 ± 0.13. As the sickness spread in the herd, the average herd entropy decreased and the associated confidence interval widened. With an increasing number of estrus cows, the herd entropy slightly increased compared to the normal herd entropy, the confidence limits of the herd entropy varied within the confidence limits observed for the normal herd and also became narrower. Furthermore, as a consequence of increasing the number of cows in estrus in the herd, there will be many high importance animals. Therefore, the variability in herd entropy will be smaller (i.e., more stable) and so will the individual AIm. This is the reason for obtaining narrower confidence interval for the herd entropy as more cows came to estrus in Fig. 11(a). Hence, examining the variability in herd entropy and individual AIm over a period of time can also be used to generate early alerts about the cows in or approaching estrus.

Fig. 11(b) and (c) depict the variability in GMM of AIm as the number of sick and estrus animals increased in the herd, respectively. When the prevalence of sick animals in the herd increased, the GMM spread widely toward the left of the mean of the normal herd (0% sick animals) and also, the bimodal nature became more apparent. Similarly, when more animal come to estrus, the GMM represented a tendency of spreading toward the right from the mean of the nonestrus case (0% case). The possibility of dividing the herd into subgroups is increasing with increasing sick/estrus animals in the herd. That is why the multimodel nature of GMM is increasing in both Fig. 11(b) and (c). Moreover, the disperse (uncertainty) of GMMs is higher with the prevalence of sickness in the herd than the increment of cows in estrus.

Therefore, exploring the most prominent features in the NN frequency matrix and the variability (or deviations) in herd

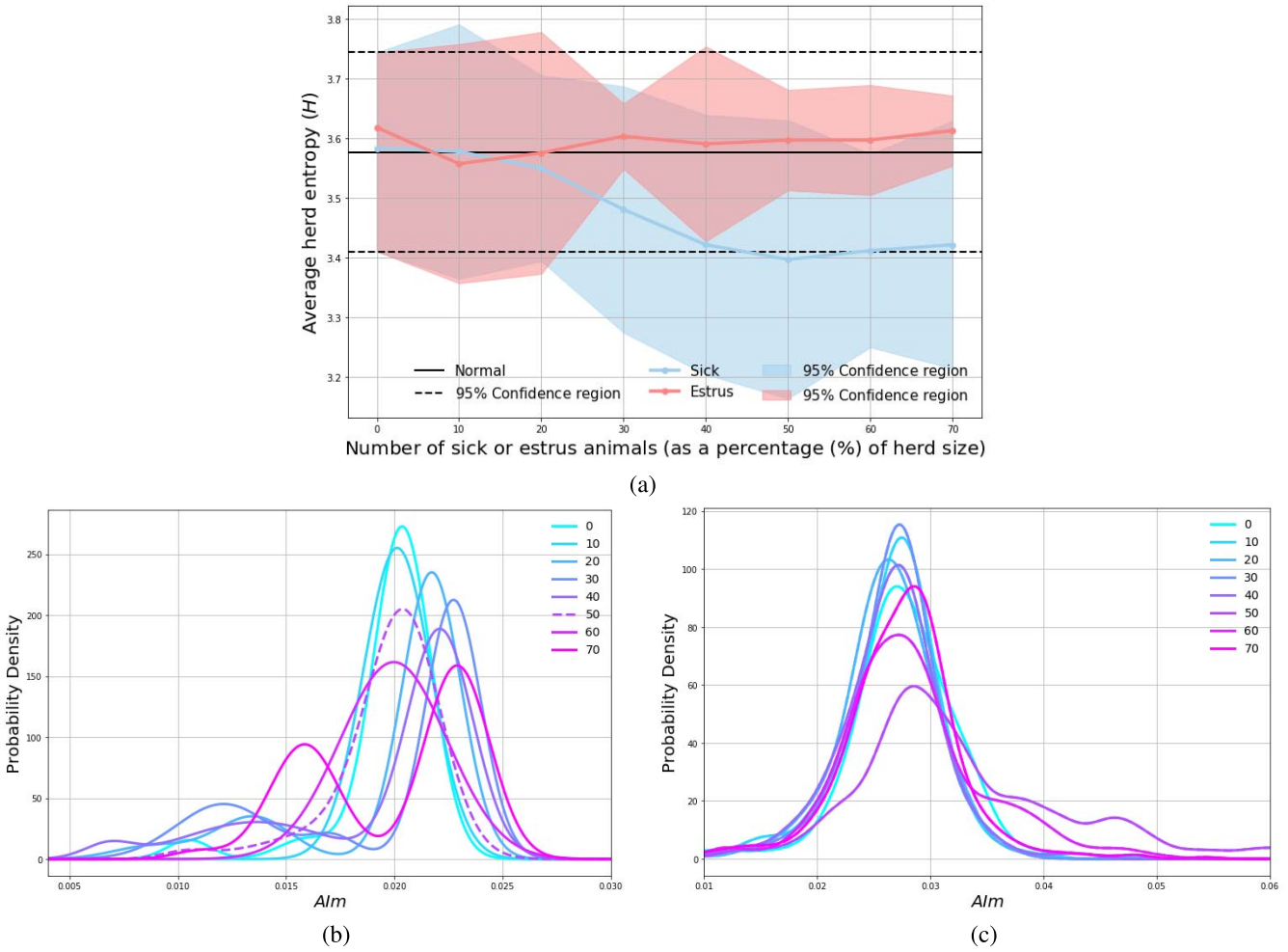


Fig. 11. Herd entropy variation with the sick and estrus animals accounted as a percentage of (a) herd size and corresponding variation of GMM for the (b) sick and (c) estrus herds.

entropy, individual PDF of AIm, as well as the GMM of AIm over a time scale, can be used to detect the presence of sick cows which may require closer attention.

B. Directions for Future Studies

The AIm cannot, however, capture time-variant properties of dynamic networks because C_W used to derive AIm and most of the other graph-theoretic measures have been defined for static graphs [18], [40]. Therefore, novel measures for dynamic network graphs are highly desirable in SNA. Moreover, animal interactions are heterogeneous in nature, and also, there are various factors which influence the generating strong interactions such as gender, age, and parental attraction. Therefore, computing interaction weights by taking into account such factors would definitely contribute to capture more precise social-behavioral characteristics. Moreover, taking into consideration more mobility parameters such as the probability for deciding the next movement direction and using the current location information when deciding the next location would help to simulate more realistic mobility data and so would the social behaviors.

In modern-day data analytic applications, interoperability of heterogeneous data sets collected across a geographically distributed data source is one of the critical issues. The reason for that is most of the technologies currently in use, notably in agrisector, operate in isolation as they are incapable of communicating with each other. Consequently, the importance of collecting data from those technologies is significantly underutilized. To make full use of such data, it is necessary to explore the interrelationships between such data sets to conduct a more comprehensive analysis. The use of AIm facilitates incorporating location-based mobility behaviors with a variety of other information sources. For instance, exploring the impact of cow mobility on milk and pasture quality, feed intake, and nutrient deficiencies. Therefore, exploring the interrelationship of AIm with other farming variables would be another exciting extension of this paper that would help alleviating the barriers of performing cooperative analytics with heterogeneous data sets.

In addition, AIm evaluating process could also be used in other applications such as WSN and distributed data analytic platforms, which are currently used in real-time decision making. In such applications, optimizing the computational, communication, and energy requirements are critical in order to improve the responsiveness and timeliness of the system. In this case, identifying the least and highest AIm nodes

can make a significant contribution. For instance, while the high AIm nodes are vital in selecting nodes (gateways) to improve the effectiveness of data communication in the network, least AIm nodes would be the best nodes for offloading computations and sharing resource requirements. Moreover, in federated learning, which is a distributed machine learning framework, a machine learning model is trained across a large number of data sources, and the model updates are aggregated at a coordinating unit to compute the final model updates. Those updates are then sent back to each data source to make inferences. Therefore, the AIm metric can be used to select the coordinating unit as the data source which has the highest AIm as such a node can effectively communicate with other data sources. Therefore, the responsiveness of the learning system can effectively be increased, optimizing the resource consumption.

VI. Conclusion

The NN-based graph analytic techniques were used in this paper to evaluate animal importance to the group, combining animal- and group-level information. Meanwhile, animal interaction range and NN frequency matrix were derived to support the AIm evaluation process. Based on the observations from the cow sample population in this paper, the most common interaction range of grazing dairy cows is 7 and the NN-frequency matrix gives an overview regarding the social affiliation of animals in the herd. The AIm metric derived based on the herd entropy variation to quantify AIm performed well compared to the already existing AIm measures, which are based on the degree and Laplacian centrality measures. Using simulated data, the intensity of cohesion in cows in estrus was greater among themselves than their cohesion to the herd. The sick cows demonstrated considerably less intensity of cohesion to the herd than healthy cows and that was even smaller compared to the cows in estrus. The PDFs of cows in estrus were shifted to the right of the mean of the PDFs of nonestrus cows, which exhibited nearly a similar mean, but differences existed in their variances. Also, cows in estrus had smaller variances (i.e., more dense PDFs) than their nonestrus contemporaries. The mean values of the PDFs of sick cows were not comparable to each other and were less than the healthy cows. Moreover, while the joint PDF of AIm in sick herd represented a bimodal nature and expanded toward the left compared to the joint PDF of AIm of a healthy herd. The joint PDF of on estrus herd showed completely an opposite behavior to the sick herd. Results from this paper, therefore, suggests that the NN frequency matrix and entropy-based animal importance metric can be used to generate early alerts about the deviations of their social behaviors and then to derive useful information.

Acknowledgment

The authors would like to thank D. McSweeney for his contribution in the collection of the global positioning system data. They would also like to thank Dr. B. Butler for his valuable comments on improving the technical quality of this paper.

References

[1] N. K. Boyland. (2018). *The Influence of Social Networks on Welfare and Productivity in Dairy Cattle*. [Online]. Available: https://ore.exeter.ac.uk/repository/bitstream/handle/10871/19360/BoylandN.pdf?sequence=1&isAllowed=y

[2] N. K. Boyland, D. T. Mlynski, R. James, L. J. N. Brent, and D. P. Croft, "The social network structure of a dynamic group of dairy cows: From individual to group level patterns," *J. Appl. Animal Behav. Sci.*, vol. 174, pp. 1–10, Jan. 2016.

[3] E. A. Codling and N. W. Bode, "Copycat dynamics in leaderless animal group navigation," *J. Moving Ecology*, vol. 2, no. 1, p. 11, 2014.

[4] R. N. Handcock *et al.*, "Monitoring animal behaviour and environmental interactions using wireless sensor networks, GPS collars and satellite remote sensing," *J. Sensors*, vol. 9, no. 5, pp. 3586–3600, 2009.

[5] B. C. Fargas and M. N. Petersen, "GPS-free geolocation using LoRa in low-power WANs," in *Proc. IEEE Int. Conf. Global Internet Things Summit (GIoTS)*, Jun. 2017, pp. 1–6.

[6] P. Varun, W. Elmannai, and K. Elleithy, "Mobile and Wi-Fi Geo location using Google latitude," in *Proc. IEEE Int. Conf. Comput. Inf. Technol. (WCCIT)*, Sousse, Tunisia, Jun. 2013, pp. 1–2.

[7] A. Goswami and A. Kumar, "Challenges in the analysis of online social networks: A data collection tool perspective," *J. Wireless Pers. Commun.*, vol. 97, no. 3, pp. 4015–4061, 2017.

[8] E. Pennisi, "Social animals prove their smarts," *Science*, vol. 312, no. 5781, pp. 1734–1738, 2006.

[9] T. Slotyasik and Z. Nogalski, "The effects of social hierarchy in a dairy cattle herd on milk yield," *Political J. Natural Sci.*, vol. 25, no. 1, pp. 22–30, 2010.

[10] M. Ballerini *et al.*, "Interaction ruling animal collective behavior depends on topological rather than metric distance: Evidence from a field study," *Proc. Nat. Acad. Sci.*, vol. 105, no. 4, pp. 1232–1237, 2008.

[11] A. Cavanga, I. Giardina, A. Orlandi, G. Parisi, and A. Procaccini, "The STARFLAG handbook on collective animal behaviour: Part II, three-dimensional analysis," *J. Animal Sci.*, vol. 76, pp. 238–248, Feb. 2008.

[12] T. Wey, D. T. Blumstein, W. Shen, and F. Jordan, "Social network analysis of animal behaviour: A promising tool for the study of sociality," *J. Animal Behav.*, vol. 75, pp. 333–344, Feb. 2008.

[13] B. A. Wood, H. T. Blair, D. I. Gray, P. D. Kemp, P. R. Kenyon, and S. T. Morris, "Agricultural science in the wild: A social network analysis of farmer knowledge exchange," *PLoS ONE*, vol. 9, no. 8, 2014, Art. no. e105203.

[14] N. Pinter-Wollman *et al.*, "The dynamics of animal social networks: Analytical, conceptual, and theoretical advances," *J. Behav. Ecol.*, vol. 25, pp. 242–255, 2014.

[15] X. Qi *et al.*, "Terrorist networks, network energy and node removal: A new measure of centrality based on laplacian energy," *J. Social Netw.*, vol. 2, no. 1, pp. 19–31, 2013.

[16] X. Ai, "Node importance of ranking of complex networks with entropy variation," *J. Entropy*, vol. 19, no. 7, p. 303, 2017.

[17] K. Park and A. Yilmaz, "A social network analysis approach to analyze road networks," in *Proc. ASPRS Annu. Conf.*, San Diego, CA, USA, 2010, pp. 1–6.

[18] K. Lerman, R. Ghosh, and J. H. Kang, "Centrality metric for dynamic networks," in *Proc. 8th Workshop Mining Learn. Graphs (MLG)*, New York, NY, USA, 2014, pp. 70–77.

[19] G. Martino, F. M. Sarti, and F. Panella, "Social network analysis in encouraging role-players in the beef market to take breeding decisions: A methodological study," *Italian J. Animal Sci.*, vol. 12, no. 1, p. e9, 2013.

[20] S. A. Rands, "Nearest-neighbour clusters as a novel technique for assessing group associations," *Roy. Soc. Open Sci.*, vol. 2, no. 1, p. 140232, 2015.

[21] L. Candeloro, L. Savini, and A. Conte, "A new weighted degree centrality measure: The application in an animal disease epidemic," *PLoS ONE*, vol. 11, no. 11, 2016, Art. no. e0165781.

[22] H. Kim and R. Anderson, "Temporal node centrality in complex networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 85, no. 2, 2012, Art. no. 026107.

[23] C. C. DubÃƒÂ©, C. Ribble, and D. Kelton, "An analysis of the movement of dairy cattle through 2 large livestock markets in the province of Ontario, Canada," *J. Can. Veterinary*, vol. 51, no. 11, pp. 1254–1260, 2010.

[24] P. Koene and B. Ipema, "Social networks and welfare in future animal management," *J. Animals*, vol. 4, no. 1, pp. 93–118, 2014.

[25] M. Nöremar, N. Håkansson, S. S. Lewerin, A. Lindberg, and A. Jonsson, "Network analysis of cattle and pig movements in Sweden: Measures relevant for disease control and risk based surveillance," *Preventive Veterinary Med.*, vol. 99, pp. 78–90, May 2011.
[26] C. Veness. (2008). *Calculate Distance, Bearing and More Between Latitude/Longitude Points*. [Online]. Available: http://www.movable-type.co.uk/scripts/latlong.html
[27] G. Neisen, B. Wechsler, and L. Gygax, "Choice of scan-sampling intervals—An example with quantifying neighbours in dairy cows," *J. Appl. Animal Behav.*, vol. 116, pp. 134–140, Jan. 2008.
[28] K. K. Nambiar, P. K. Varma, and V. Saroch, "An axiomatic definition of Shannon's entropy," *Appl. Math. Lett.*, vol. 5, no. 4, pp. 46–54, 1992.
[29] X. Qi, E. Fuller, Q. Wu, Y. Wu, and C.-Q. Zhang, "Laplacian centrality: A new centrality measure for weighted networks," *Inf. Sci.*, vol. 194, pp. 240–253, Jul. 2012.
[30] D. Reynolds, "Gaussian mixture models," MIT Lincoln Lab., Lexington, MA, USA, Tech. Rep., 2007. doi: 10.1007/978-1-4899-7488-4_196.
[31] A. Bach and V. Cabrera, "Robotic milking: Feeding strategies and economic returns," *J. Dairy Sci.*, vol. 100, pp. 7720–7728, Sep. 2017.
[32] K. Gajamannage, E. M. Bollt, M. A. Porter, and M. S. Dawkins, "Modeling the lowest-cost splitting of a herd of cows by optimizing a cost function," *J. Nonlinear Sci.*, vol. 27, no. 6, 2017, Art. no. 063114.
[33] H. Kim, S. Oh, S. Ahn, and B. Choi, "Real-time temperature monitoring to enhance estrus detection in cattle utilizing ingestible bio-sensors: Method and case studies," *J. KIIT*, vol. 15, no. 1, pp. 65–75, 2017.
[34] A. Panisson. (2012). *Generalized Random Waypoint to Support any Number of Spatial Dimension*. [Online]. Available: https://github.com/panisson/pymobility
[35] P. M. Fricke, P. D. Carvalho, J. O. Giordano, A. Valenza, G. J. opes, and M. C. Amundson, "Expression and detection of estrus in dairy cows: The role of new technologies," *J. Animal*, vol. 8, no. s1, pp. 134–143, 2014.
[36] J. B. Roelofs, F. J. C. M. van Eerdenburg, N. M. Soede, and B. Kemp, "Various behavioral signs of estrous and their relationship with time of ovulation in dairy cattle," *J. Theriogenology*, vol. 63, no. 5, pp. 1366–1377, 2005.
[37] M. L. O'Connor. (2017). *Heat Detection and Timing of Insemination for Cattle*. Accessed: Nov. 2017. [Online]. Available: https://extension.psu.edu/heat-detection-and-timing-of-insemination-for-Cattle
[38] S. L. Walker, R. F. Smith, J. F. Routly, D. N. Jones, M. J. Morris, and H. Dobson, "Lameness, activity time-budgets, and estrus expression in dairy cattle," *J. Dairy Sci.*, vol. 91, no. 12, pp. 4552–4559, 2008.
[39] A. Rahaman, D. Smith, J. Hills, G. Bishop-Hurley, D. Henry, and R. Rawnsley, "A comparison of autoencoder and statistical features for cattle behaviour classification," in *Proc. Neural Netw. (IJCNN)*, Vancouver, BC, Canada, Jul. 2016, pp. 1232–1237.
[40] A. J. Itle, J. M. Huzzey, D. M. Weary, and M. A. G. von Keyserlingk, "Clinical ketosis and standing behavior in transition cows," *J. Dairy Sci.*, vol. 98, no. 1, pp. 128–134, 2015.
[41] M. S. Herskin, F. Skjøth, and M. B. Jensen, "Effects of hunger level and tube diameter on the feeding behavior of teat-fed dairy calves," *J. Dairy Sci.*, vol. 93, no. 5, pp. 2053–2059, 2010.

Dixon Vimalajeewa received the B.Sc. degree in mathematics and statistics from the University of Ruhuna, Matara, Sri Lanka, in 2012, and the M.Sc. degree in computational engineering from the Lappeenranta University of Technology, Lappeenranta, Finland, in 2015. He is currently pursuing the Ph.D. degree with the Telecommunications Software and Systems Group, Waterford Institute of Technology, Waterford, Ireland.

His current research interests include data analytics, sensor-based animal phenotypes, and distributed learning algorithms.

Sasitharan Balasubramaniam received the bachelor's degree in electrical and electronic engineering from The University of Queensland, Brisbane, QLD, Australia, in 1998, the master's degree in computer and communication engineering from the Queensland University of Technology, Brisbane, in 1999, and the Ph.D. degree from The University of Queensland in 2005.

He is currently an Academy of Finland Research Fellow with the Department of Electronic and Communication Engineering, Tampere University of Technology, Tampere, Finland, and an Acting Director of research with the Telecommunication Software and Systems Group, Waterford Institute of Technology, Waterford, Ireland, where he was involved in a number of Science Foundation Ireland projects. His current research interests include molecular and nanocommunications and Internet-of-(bio–nano) Things.

Dr. Balasubramaniam is the Co-Founder of the Steering Committee of the ACM NanoCom Conference. He was a recipient of the ACM/IEEE NanoCom Outstanding Milestone Award in 2018. He is the IEEE Nanotechnology Council Distinguished Lecturer. He is currently an Editor of the IEEE INTERNET OF THINGS JOURNAL, *Nano Communication Networks* (Elsevier), and *Digital Communication Networks*.

Chamil Kulatunga received the bachelor's degree in electronics and telecommunication engineering from the University of Moratuwa, Moratuwa, Sri Lanka, in 1999, the master's degree in computer science from the Waterford Institute of Technology (WIT), Waterford, Ireland, in 2003, and the Ph.D. degree in Internet engineering from the University of Aberdeen, Aberdeen, U.K., in 2009.

From 2015 to 2017, he was an Experienced Post-Doctoral Researcher, under the Science Foundation Ireland funded PrecisonDairy Project, with the Telecommunication Software and Systems Group, WIT. He is currently a Research Data Analyst, under the Science Foundation Ireland and Origin Enterprises funded CONSUS Project, with the University College Dublin, Dublin, Ireland. His current research interests include distributed machine learning and data-driven agriculture in crop and dairy farming.

Bernadette O'Brien received the bachelor's degree in dairy science and the master's and Ph.D. degrees from the National University of Ireland, Cork, Ireland, in 1982, 1985, and 1988, respectively.

She is currently a Principal Research Officer with Teagasc Moorepark's Animal and Grassland Research and Innovation Centre, Cork. In her current role at Teagasc, her projects include research into innovative and sustainable systems combining automatic milking and precision grazing, increasing efficiency of traditional technologies, and exploring new technology on dairy farms. This has followed from research on on-farm labor efficiency with respect to labor input profiles, alternative milking frequencies, various calf feeding frequencies, and milking efficiency in conventional parlors. She has also been successful in obtaining funding for research initiatives examining the use of precision technologies, technology platforms, and computational biology to increase the economic and environmental sustainability of pasture-based production systems. She has recently established and coordinated a network of research scientists in six European countries in joint research in automatic milking.

Donagh P. Berry received the bachelor's degree in agricultural science and the Ph.D. degree in quantitative genetics from the University College Dublin, Dublin, Ireland in 2000 and 2003, respectively, and the master's degree in bioinformatics and systems biology from the University College Cork, Cork, Ireland, in 2012.

He is currently a Senior Principal Investigator of quantitative geneticist with Teagasc, Cork, Ireland, where he is responsible for the research on genetics in dairy cattle and for the development and implementation of genomic evaluations in dairy cattle, beef cattle, and sheep in Ireland, and the Director of the VistaMilk Agri-Tech Research Centre, where he leads a team of more than 200 scientists in the development and deployment of digital technologies in precision dairy production. He holds professorships at three (inter)national universities.

Appendix D

A Service-based Joint Model Used for Distributed Learning: Application for Smart Agriculture

Journal Title:	IEEE Transactions on Services Computing
Article Type	Regular Paper
Complete Author List	Dixon Vimalajeewa, Chamil Kulatunga, Donagh P. Berry, Sasitharan Balasubramaniam
Status	Under Review

A Service-based Joint Model Used for Distributed Learning: Application for Smart Agriculture

Dixon Vimalajeewa, Chamil Kulatunga, Donagh P. Berry, Sasitharan Balasubramaniam

Abstract—Advanced distributed analytics facilitate to make the services smarter for a wider range of data-driven applications in many domains, including agriculture. The key to producing services at such level is timely analysis for deriving insights from data. Centralized data analytic services are becoming infeasible due to limitations in both the ICT infrastructure, timeliness of the information, and data ownership. Distributed Machine Learning (DML) platforms facilitate efficient data analysis and overcome such limitations effectively. Federated Learning (FL) is a DML concept, enables optimizing resource consumption while performing privacy-preserved timely analytics. In order to create such services through FL, there needs to be innovative machine learning (ML) models as data complexity as well as application requirements limit the applicability of existing ML models. Therefore, in this paper, we propose a Neural Network (NN)- and Partial Least Square (PLS) regression- based joint FL model (FL-NNPLS). Then its predictive performance is evaluated under sequential- and parallel-updating based FL in a smart farming context, and specifically for milk quality analysis. Smart farming is a fast-growing industrial sector which requires effective analytic platforms to employ sustainable farming practices. The FL-NNPLS approach performs and compares well with a centralized approach and has state-of-the-art performance.

Index Terms—Decentralized Machine Learning, Federated Optimization, Neural Network, Data Imbalance, MIRS Milk Quality Predictions.

1 Introduction

Extensive adoption of connected technologies such as Internet of Things (IoT) and Cloud Computing are propelling the advancement of data-driven services in many sectors such as sustainable intensification of food production in agriculture and for personalised/controlled delivery of drugs in health-care. To deliver such services, data analytic frameworks are essentially required to be integrated to the ICT infrastructures to extracting insights and then communicate them effectively to the consumers and end-users for. The reason for this is such integrated systems can handle heterogeneous and massive datasets while enabling granular analysis with dynamic changes to produce timely and accurate insights. Hence, an advanced analytical platform coupled with efficient Machine Learning (ML) techniques is required to improve reliability and timeliness of such services [1], [2], [3]. Therefore, this study focuses on proposing an effective analytical framework based on Distributed ML (DML) models coupled with IoT and Cloud Computing infrastructures.

Most sensor technologies and IoT platforms provide services to collate and store vast quantities of data produced from geographically distributed sources. As most computational facilities for analyzing such data reside in centralized data centers (e.g., cloud), where data will be consolidated as single large datasets, means that the analytics can subsequently be performed at the location, which is referred to as Centralized ML (CML). Numerous studies have shown that CML is highly advantageous for developing new hypothesis, as it enables improved learning accuracy [4] and model acceptability [5]. However, data centralization is feasible only when the communication and computational capabilities of the data centers are not limited. At the same time, if data owners are not reluctant to sharing data with CML systems due to data privacy, security, and ownership concerns [6]. These limitations hinder real-time decision making, which is crucial in providing timely services. A promising approach to solve this is Distributed ML (DML), as it facilitates the development of more advanced intelligent systems by incorporating various systems, technologies, and ML techniques. Therefore, to overcome the limitations in CML, there is a growing need for effective DML approaches equipped with functionalities such as data-protection and optimum use of available resources.

DML synchronously or asynchronously executes data-intensive analytical applications across geographically distributed processing units by capturing real-time dynamics in order to enable making timely decisions for supporting services effectively [7]. For instance, Fog computing paradigm is emerging as a technological enabler for DML, since the analytical process is discovered and offloaded to a

- *D. Vimalajeewa is with the Telecommunications Software and Systems Group, Waterford Institute of Technology, Waterford, Ireland, E-mail: dvimalajeewa@tssg.org.*
- *C. Kulatunga is with the University College Dublin, Ireland, E-mail : chamilkul@gmail.com.*
- *D. P. Berry is with the Teagasc, Animal & Grassland Research and Innovation Centre, Moorepark, Fermoy, Co. Cork, Ireland, E-mail : Donagh.Berry@teagasc.ie.*
- *S. Balasubramaniam is with Telecommunications Software and Systems Group, Waterford Institute of Technology, Waterford, Ireland and Faculty of Information Technology and Communication Sciences, Tampere University, Finland, E:mail sasib@tssg.org.*

This work was supported by the Science Foundation Ireland (SFI) projects PrecisionDairy (ID: 13/1A/1977) and VistaMilk (ID: 16/RC/3835), and the Horizon 2020 GenTORE project.

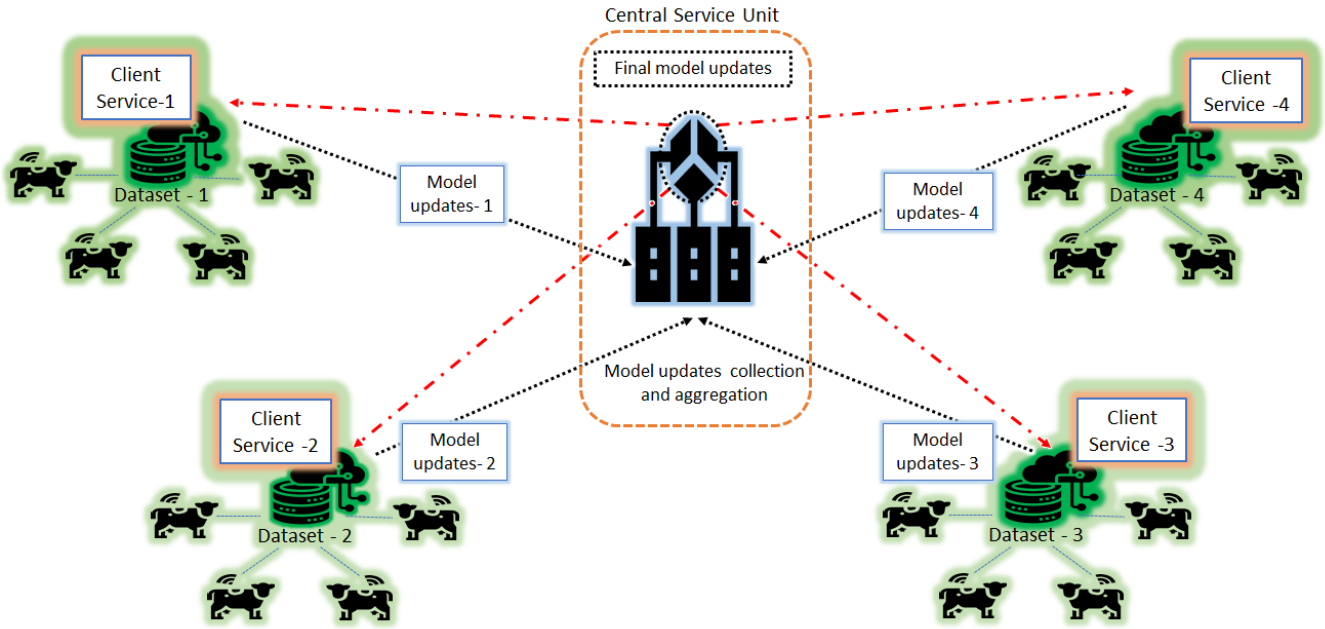


Fig. 1. A FL architecture based on smart dairy farming: (1) Every client service (i.e., farm) collects data (e.x., MIRS of milk samples) and trains a ML model and send the updated model to the central service unit, (2) The central service unit which derives the final model updates by aggregating the model updates of client services, (3) Client services download the final model updates and then update their local ML models to perform inferences.

node that is in close proximity to the data source [8]. This has led to a growing interest to focus on developing new distributed optimization techniques [9] and one technique is Federated Learning (FL) [10]. FL enables optimization algorithms to be federated in order to make real-time decisions without moving data away from the source. This also minimises data privacy concerns and needed resource consumption (this will be elaborated in a subsequent section).

The key to practicing such DML techniques is having effective ML models which can be trained dynamically for capturing underlying updated data. The data collected by modern day sensor technologies and IoT platforms is usually highly dynamic, complex, large in size, and highly dimensional which can lead to noise accumulation, multi-correlation, and heavy computational costs. Consequently, such characteristics limit the applicability of commonly used simple ML models such as Least Square Regression (LSQR), Principal Component Analysis (PCA), Partial Least Square Regression (PLSR), and Neural Network (NN). For example, LSQR, PLSR, and NN techniques have extensively been used in various data analytical applications, but fails under certain conditions, thus requiring further enhancements. That is, although PLSR overcomes the limitations of multi-colinearity and high-dimensionality in LSQR fitting, the predictive accuracy is limited due to its inability in capturing complex functional relationships, such as non-linearity [11]. A promising alternative to mitigate such limitations is NN models, as they are becoming increasingly feasible and also actively being used for a wide range of applications. However, selecting optimal NN configurations by using techniques such as grid search and random search depends on the problem, resource availability as well as the experience of the users. It may be inefficient in terms of time and resource consumption as mostly IoT sensor-based DML applications are resource constrained and timely delivery of the outcomes is vital. This means that the NN approach may not be an efficient alternative in all cases.

Consequently, modern studies are exploring innovative ML models which are, for instance, scalable, adaptive, light-weight, and computationally inexpensive. It has already been proven that the combined use of NN models with conventional ML models such as PLSR and PCA (known as hybrid models) has the potential to alleviate the barriers that arise when they are used separately, while providing accurate outcomes. However, this scenario has been used in a limited number of studies such as [11], [12], and also under CML settings. To date, there has been no study that has investigated its use within DML, and in particular under FL. Therefore, in this study, we propose a joint model of coupling the NN with PLSR models that can be used for FL. We demonstrated the effectiveness of FL in the context of smart dairy farming by using Mid-Infrared Spectroscopy (MIRS) analysis of milk samples, which are routinely used for the quantification of milk quality (see Fig.1). Milk quality analysis particularly plays a vital role in the dairy industry [13], where micro-nutrients of milk components monitored by robotic milking machines can be analyzed using ML techniques to determine several important insights such as health issues [14], nutrient deficiencies [15], and social behaviour [16]. These insights provide support in terms of intensifying sustainable farming practices such as controlled delivery of nutrients and fertilizers and reducing cost while increasing the profit. However, MIRS-based chemometric analysis have extensively used PCA, LSQR, and PLSR models in earlier studies such as [17], [18]. Joint ML models have not generally been used under FL settings for any analytical purposes, and particularly in the field of agriculture. Therefore, our main contributions of the study is summarized as follows:

- By considering the limitations of the LSQR, PLSR,

and NN techniques, introduce a joint ML model known as NNPLS, which combines NN and PLSR techniques.
- Evaluate two federation approaches, which are sequential and parallel, where they can be used to federate the NNPLS model, and apply these approaches to a Smart Dairy Farming application. Specifically, the application is for Mid-Infrared Spectroscopy (MIRS) milk quality data analysis.
- Evaluate and compare the predictive performances of the NNPLS model under CML and DML (FL) settings.
- Ensure that the NNPLS model based FL has comparable performance to the state-of-the-art methods by comparing FL performance of the NNPLS model to an advanced ML technique, which is Convolutional NN (CNN) model.
- Discuss the advantages and challenges of the NNPLS-based FL while proposing an alternative to overcome the imbalance issues in FL.

The remainder of the paper is organnized as follows. Section 2 summarizes the state-of-the-art in distributed machine learning, including the discussion of the FL process followed by the NNPLS model. FL approaches are discussed in detail in Section 3. Section 4 presents the performances of NNPLS under FL settings while Section 5 and 6 discusses the benefits and limitations of the FL-NNPLS approach and concludes the paper, respectively.

2 State-of-the-art on DML and FL

The progressive advancements of IoT and cloud technologies have contributed to development of smarter services that are found in various sectors such as transportation [19], healthcare [20], agriculture [21], [22], and energy [23]. Therefore, this section provides an overview of the relevant literature focusing on the attempts made in advancing DML mechanisms, including the importance of FL in DML, and the involvement of deep learning and hybrid ML models in ML.

2.1 Distributed Machine Learning (DML)

CML has traditionally been the dominant ML approach. However, with the growing prevalence of big data, CML approaches are facing increasing challenges in collecting and processing massive datasets due to constraints in resource for modern ICT platforms [6], and in particular low powered devices. Consequently, the concept of DML is gaining traction for a wide range of applications such as image classification [9], smart healthcare [20], and smart agriculture [21]. This has led to continued development in new DML algorithms with high accuracy and fast convergence rates [9]. As a result, most of the CML algorithms were converted into DML models, e.g., Bayesian networks, decision trees, and support vector machine [24]. Nevertheless, to support the ever-increasing complexity of data and their features, research in ML has focused on incorporating learning functionalities, and one example is Deep Learning [6]. In employing learning functionalities in a broader spectrum of DML applications, data-protection was raised as one of the major concerns and was even more severe in large-scale data analytic applications [21], [24]. As a result, privacy-preserved DML solutions such as FL [10] and parameter server-based DML [25], blockchain-enabled ML [26] have been developed. When there is a trusted third party which can serve as an central service unit, FL and parameter server-based approaches were recommended for DML. Other alternative is the use of blockchain but discovering resources providers was the most critical challenge.

Considering the early attempts that have been made to carry out DML, the MapReduce computational model proposed in [27] migrated computation towards the source of the data, which significantly reduced the communication requirements of big datasets, even when Hadoop stored data in commodity hardware clusters (sometimes geographically distributed) and processed data locally in batches [28]. The parameter server-based DML proposed in [25] performs asynchronous data communication between distributed nodes, supporting flexible consistency models, elastic scalability, and continuous fault tolerance. Today, the Fog and Edge computing paradigms are emerging as the best enablers for DML algorithms. The reason is that analytics in Fog and Edge computing provides opportunities for offloading centralized computations by discovering processing capabilities in close proximity to the data sources [7].

2.2 Federated Learning (FL)

FL is a collaborative ML (and also a distributed optimization) concept [29] that was developed by Google researchers to train distributed datasets in a centralized server [25]. A shared predictive model is collaboratively trained without moving data away from the participating client service's nodes such as mobiles and gateway devices. FL process mainly consists of three steps, which are as follows (also see Fig. 1):

1. Each client service trains (updates) the model, i.e., computes the model parameters by using its own data, and then transmits updated models (local models) to the central service unit.
2. The central service (coordination) unit (e.g., parameter server) collects the local models (i.e., local model parameters) and then aggregates them to compute the final updated global model parameters. That is, every client services contributes to collaboratively train a common ML model.
3. Every client service retrieves global model parameters from the central service unit and uses them for making their own decisions and also, for the next model updating cycle.

These three steps repeatedly execute as FL progresses.

Gradient Descent (GD) algorithms have commonly been used for computing model parameters [30]. Assume a FL model that needs to update a parameter matrix of $W^{d_1 \times d_2}$. The i^{th} client service (assume C number of client services in total) will download the parameters at time t, W_t^i, and calculate the gradient $H^i = W^i - W$. Then, $W_{t+1} = W_t + \eta H_t$, where η is the learning rate. According to the average principle, $H_t = \frac{1}{n} \sum H_t$. There are some variants of GD algorithms such as mini-batch GD and Stochastic GD(SGD) [31].

In SGD, a single training sample is used at a time to train the model while mini-batch GD uses a small subset of the training dataset. Both have been proposed to overcome the processing burden in deep learning algorithms using image data with a vast number of parameters in the order of 1000s or more.

Structured- and sketched-updating are two main approaches used to optimize the up-link communication for sending the updated parameters to the server when the resources are restricted. The structured approach sends the full gradient to the central procession unit while the sketched approach transmits only the compressed gradient. These two approaches are, however, used together in FL and the federation process can be realized by following the synchronous or asynchronous FL algorithm. In the asynchronous algorithms, all the training subsets are updated by the client services independently, while in the case of synchronous case, the parameter updates are dependant between all the client services. Synchronous algorithms are comparatively slow since the lower bound delays is constrained by the most inefficient machine in the cluster of client services while the asynchronous approach is known to converge towards poorer results [29].

The quality of the federated ML model and the optimization technique which is used to train the model are as crucial as the quality of the data used. Since a CML model is transformed into a set of small devices, each using a subset of a large training dataset, this model achieves data locality and harnesses the computational power of distributed edge/fog computing. Also, FL makes use of idle processing power by facilitating a certain number of clients to act as model trainers. Since the datasets are located in the client services' devices, FL has great potential in reducing communication cost and preserves the data ownership and privacy issues. Recent research [20], which used FL with the support vector machine classifier for predicting heart diseases, and preserves the privacy of patients data, is good evidence to emphasize the significance of FL. Moreover, the reasons to use FL as an alternative to current ML techniques with privacy preserving were broadly discussed in [10], while highlighting some common issues of FL, such as data imbalance and misbehaviour (or failure) of the central service unit, which requires careful attention when selecting it.

2.3 Involvement of Deep Learning in modern ML

The involvement of deep learning techniques such as convolution, recursive, and recurrent NNs in advanced ML research like image recognition, object detection, and video analysis [11], [32] has intensified in recent years. Consequently, FL has also been influenced by deep learning techniques. For instance, the study in [30] used a CNN model to study the communication efficiency of their ML model under FL settings. The local Stochastic Gradient Descent (SGD) optimization technique based *Federated Averaging* algorithm proposed in [30] proved that the ML model could train with minimal communication requirements and that could help effectively overcome the communication issues in FL [10]. However, the applicability of deep learning models was limited particularly in many resource-limited applications such as sensor-based analytics.

As an alternative, hybrid ML models, i.e., the combined use of deep learning models with conventional ML models, were developed. For instance, the study [12] proved that the combined use of NN and PLSR has the potential to generate more robust outcomes as well as optimize the resource consumption compared to using the algorithms separately. Similarly, different versions of hybrid models such as PCA combined with NN (PCANet) [33], PLSR with NN (PLSNet) [34], CNN based PLSR (CNN-PLS or stacked PLS) [32], and Hybrid-DBSCAN model for optimizing clustering throughput in GPU and CPU [35] have been proposed and applied in various applications. Some of the applications include failure diagnosis of railway infrastructures [12], traffic incident detection [11], inland water quality evaluation [36], and image analysis [37]. The use of hybrid model was always limited to deep learning applications and has been realized in many other purposes as well. For example, [38] proposed a hybrid mathematical formulation for optimizing the computational energy required at data centers intending to reduce the ecological impact arising from data processing (i.e., Greenhouse gas emission). Also, [39] proposed a hybrid energy harvesting system and proved that the proposed system has greater performances when compared to a single source harvesting approach. However, there is no evidence that such hybrid models have been used under DML settings, and in particular for FL framework.

2.4 Smart Dairy Farming

With the growing adoption of modern technologies such as milking robots and remote sensing in smart farming, more data-driven and data-enabled services are available today. Timely recommendations and relevant management strategies based on analyzing the information collected play a crucial role in accelerating the sustainable intensification of food production while optimizing resource utilization [21], [40]. This is a major requirement to address the challenges that will result from limited land availability, high labour cost, as well as climate change, aiming to support food demand for the 9 billion world population by 2050 [41]. This not only emphasises the need for effective computing services equipped with ML mechanisms that are up-to-date and trained dynamically, but have the capacity to be distributed and cooperative [40], [42]. However, most farms operate in isolation, which in turn, limits their interoperability. Resource constraints, functional incompatibility of the existing IoT platforms hamper such analyses. Besides these limitations, farmers are reluctant to share their data due to privacy and ownership issues. In this context, FL can provide effective services for deriving decisions by integrating insights extracted from a large number of distributed data sources.

In general, FL has been beneficial for supporting services for different applications and particularly where data privacy, constraints in resources, and ownership becomes major concerns in performing data analytics. At the same time, the use of hybrid ML models in data analytics have received considerable attention for a wide range of data analytical applications. The importance of FL, however, has not been thoroughly realized yet in smart agriculture where communication and computation resource limitations and also data-privacy are significant issues.

3 Federated Learning with a Neural Network based Partial Least Square Regression

This section, first, briefly describes the NN and PLSR methods and then explains how these two methods are combined to derive the joint ML model known as NNPLS. Next, the two approaches to federate the derived joint models are discussed. Finally, the evaluation metrics used for assessing the FL performances are described.

3.1 Partial Least Square Regression (PLSR)

Least square regression (LSQR) fails when the predictor variables are strongly correlated with each other and the number of informative features is larger than the number of data points. PLSR is a projection method [43], and considers not only the correlations between the predictor variables (X), but also the correlations among the predictor and the response (y) variables. By doing so, PLSR overcomes the limitations of the LSQR method, transforming the dataset into a lower dimensional space (latent space), where the LSQR can be used. Therefore, the general procedure of PLSR consists of two steps; dimension reduction and the application of LSQR, and are listed as follows:

1. The PCA technique is used for the dimension reduction by deriving the PLS factors (or Latent Variables (LVs)), which explain most of the variation in X and y.
   That is, PCA decomposes X and y using the singular value decomposition method as: $X = G_X P_X^T$ and $y = G_y P_y^T$, where G and P represents the score and loading matrices (their subscript stands for the matrix which they are derived from), respectively. Suppose q LVs are selected, where normally cross-validation technique is used) the X can be represented as:
   $$X = g_{X,1}p_{X,1}^T + g_{X,2}p_{X,2}^T + \cdots + g_{X,q}p_{X,q}^T + E_{X,q}, \quad (1)$$
   where $\{g_{X,i}\}_{i=1}^q \in G_X$, $\{p_{X,i}^T\}_{i=1}^q \in P_X$, and $E_{X,q}$ is the error matrix when the first k LVs are used to form the PLSR model.
2. LSQR is used to derive the PLSR model as follows,
   $$y = p_{X,1}^T p_{y,1}^T + p_{X,2}^T p_{y,2}^T + \cdots + p_{X,q}^T p_{y,q}^T + E_{y,q} \quad (2)$$
   where $\{p_{y,i}\}_{i=1}^q \in P_y$ and $E_{y,q}$ is the error vector.

Further details on the PLSR process can be found in [43].

3.2 Neural Network (NN)

NN is the neural structure of the human brain, which is used for non-linear computational models with self-learning features. The model is constructed from interconnected layers of nodes that represents artificial neurons. The nodes of consecutive layers are connected by weighted links, which communicate information between the layers. The incoming data to these nodes are processed using a function called activation function. The data fed into a NN via the input layer are processed through the hidden layers, and the outcomes are derived from the output layer (Fig. 2). A learning rule is used to adjust the weights of the links to minimize the errors of the learning outcomes (e.g., back-propagation technique).

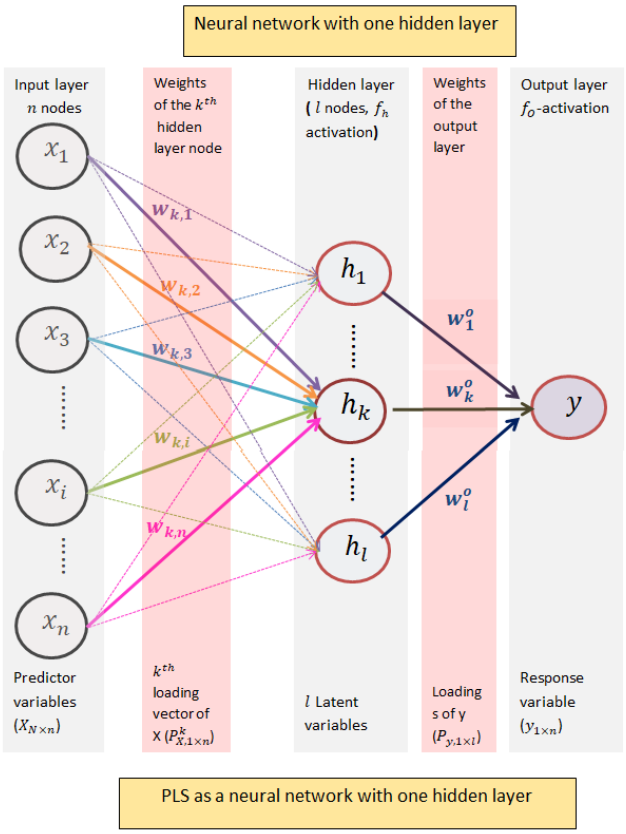


Fig. 2. PLSR model as a naive NN with one hidden layer. The top and bottom of each vertical color bar explains the entities of the NN and PLSR models, respectively. The weights in the hidden layer corresponds to the k^{th} node/LV.

We assume a simple NN with n nodes in the input layer with a single hidden layer, which consists of l nodes, and the output layer has only one node (see Fig. 2) and named as $(n, l, 1)$ NN model. $\{W_{n\times l}^h, b_{1\times l}^h\}$ and $\{W_{1\times l}^o, b^o\}$ are respectively the input and hidden layer weight (W) matrices, including the bias (b) term. The incoming and the outgoing information, denoted as h_{in}^t and h_{out}^t of the k^{th} hidden layer nodes are computed as follows (Fig. 2 provides the description of the process at the top of each colored bars):

$$\begin{aligned} h_k^{in} &= w_{k,0} + x_1 w_{k,1}^h + x_2 w_{k,2}^h + \cdots + x_n w_{k,n}^h \quad (3) \\ &= \sum_{i=0}^{n} x_i w_{k,i}^h \\ h_k^{out} &= f_h(h_k^{in}), \quad (4) \end{aligned}$$

where f_h is the hidden layer activation function, $\{w_{k,i}\}_{i=1}^l$ is the k^{th} column of the matrix W^h, and $w_{k,0}$ is the k^{th} element of the vector b^h. The input data sample is represented as $x_1, x_2, \cdots, x_n \in X_{N\times n}$.
The same procedure is repeated for all other nodes in order to compute their outputs that will represent the inputs for the output layer (y_{in}). Based on this, NN calculates the

output (y_{out}) as follows:

$$y_{in} = w_0^o + h_1^{out} w_1^o + h_2^{out} w_2^o + \cdots + h_l^{out} w_l^o \quad (5)$$

$$= \sum_{i=0}^{l} h_i^{out} w_i^o \quad (6)$$

$$y_{out} = f_o(y_{in})$$

where $w_0^o = b^o$ and $\{w_i^0\}_{i=1}^l \in W^o$ are the output layer bias and weight matrix, respectively, and f_o is the output layer activation function.
In order to improve the robustness of y_{out}, NN optimizes the weights as well as the bias by minimizing the errors between the actual y and y_{out}. This is referred to as training the NN. The most extensively used training algorithm is the back-propagation and the process explained above presents the forward propagation step as it required for explaining the process of deriving the joint model only. Whereas, the back-propagation technique is used for training the joint model. There are other different optimization techniques for learning rules such as SGD. $f(x) = x$ (*linear*), $f(x) = \frac{1}{1+e^{(-x)}}$ (*Sigmoid*) are some of the frequently used activation functions.

3.3 NNPLS Model

When the number of hidden nodes is equal to the number of LVs (i.e., $l = q$), and the input and output layer weight matrices (i.e., W^h and W^o) is equal to the loading matrix of X (P_X^T) and y (P_y^T), respectively, the computations represented by equation 3 and 5 are equal to the equation 1 and 2, respectively. In other words, one forward propagation step of the $(n, l, 1)$NN model given in Fig.2 is equal to the PLSR model with $q(= l)$ number of latent variable. On the other hand, the PLSR model can be considered as a NN model with one hidden layer (number of hidden layer nodes is equal to the number of latent variables). The single node in the output layer is illustrated in Fig 2. This is the basic concept behind the joint PLSR and NN techniques for deriving the NNPLS model. Then the derived NNPLS model is trained in four steps, which are as follows (see Fig.3):

1. Apply suitable pre-processing on a given dataset $[X, y]$ such as PCA, scaling, and centering.
2. The optimal number of hidden nodes required for the NNPLS model is the number of LVs, which is derived from the PLSR-based cross-validation technique.
    a. Different PLSR models are fitted to the data by varying the number of LVs. In each fitting, the cross-validation error ($RMSE_{CV}$ - explained later) is computed by repeating the 10-Fold CV for 10^3 iterations.
    b. The number of LVs corresponds to the minimum $RMSE_{CV}$ and is selected as the optimal number of LVs which is also the number of hidden nodes (l) for the NNPLS model. For instance, in order to get an idea about selecting optimal LVs, Fig. 4 represents the behavior of $RMSE_{CV}$ with respect to the LVs for Fat milk percentage using the MIRS dataset discussed in Section 4.1. As observed in the graph, the minimum $RMSE_{CV}$ occurs at 6 LVs, which is the number of hidden nodes required for NNPLS model.
3. The loading matrices of X and y that corresponds to the optimal LVs are taken as the initial weights of the input and output layers that are required for starting the NN training process with selected activation functions and the optimization technique. The *rectified* and *linear* activation functions were respectively used for f_h and f_o. The optimization technique used was the $ADAM$ method.
4. Perform model updating based on a preferred approach and here we explain the sequential and parallel updating under FL settings in the next section.

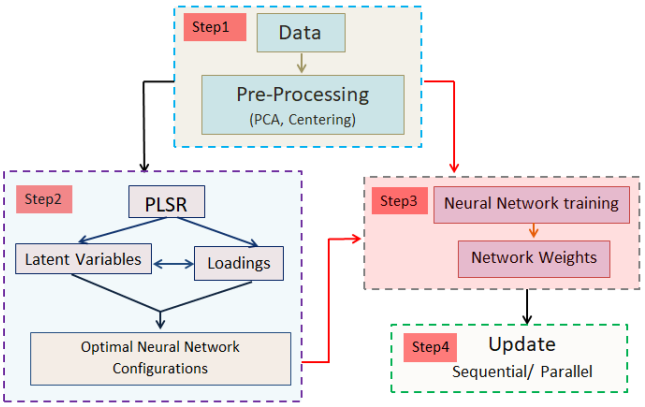


Fig. 3. Workflow of the NNPLS model for data pre-processing and model development steps.

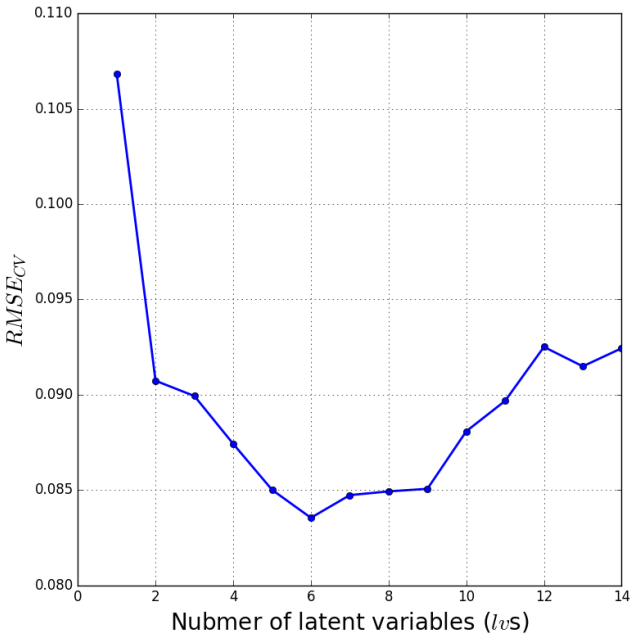


Fig. 4. Determination of the optimal number of LVs (i.e., hidden layer nodes) by using the MIRS dataset explained in Section 4.1 with Fat milk quality parameter; the optimal value of LV corresponds to the minimum $RMSE_{CV}$.

3.4 NNPLS model based Federated Learning (FL-NNPLS) architecture

In the NNPLS model based FL, the set of parameters that has to be federated is the NN weights derived from each client service that is located in distributed locations. Each client service trains a common NNPLS model by using the weights downloaded from the central service unit. After training, the client service will send back the updated weights to the central service unit. The central service unit aggregates them and computes the final updated weights, and sends back to every client service unit in order for them to update their models and perform predictions. In this process we assume that all client services contribute to model updating at each federation step as well as accept the final model updates. This means that the NNPLS model is dynamically updated based on the new datasets collected from each client over time. This is the core functionality of the FL-NNPLS architecture, which updates the model sequentially as well as in parallel.

3.4.1 Parallel Updates

The process for parallel updates follows the three steps illustrated in Fig.1, and is as follows:

1. Each client service independently trains a common NNPLS model using the available data.
2. The central service unit collects the NNPLS model weights sent by each client then averages it in order to get the final updated global weights
3. Each client service unit downloads the global weights and update the model to perform predictions accordingly.

In the next federating step, which is the updating process, the global weights from the previous federating step are combined with the new loading metrics derived from the PLSR method by using the new dataset, and this is then used as initial weights for training the NNPLS model.

Let us assume there are C client service units, the at the $(t-1)^{th}$ federating step, the final input and output NNPLS model weights are denoted as W_{in}^{t-1} and W_{out}^{t-1}, respectively, and are derived by averaging the weights received from the C client services as follows:

$$W_{in}^{t-1} = \frac{1}{C}\sum_{i=1}^{C} W_{in,i}^{t-1} \qquad W_{out}^{t-1} = \frac{1}{C}\sum_{i=1}^{C} W_{out,i}^{t-1}$$

where $W_{in,i}^{t-1}$ and $W_{out,i}^{t-1}$ are the input and output NNPLS model weight matrices of the i^{th} client.
These weights are then sent back to all the client services to assess their model's performance and is also used for computing the initial weights for the next federation step (t^{th}). At the t^{th} step, the i^{th} client service applies the PLSR to its new dataset, $\{X_i^t, Y_i^t\}$ and derives the loading matrices $P_{X,i}^t$ and $P_{Y,i}^t$. Following this process, the initial weights for training the NNPLS model are computed by averaging these loading matrices with the final model weights from the $(t-1)^{th}$ step, which is represented as follows:

$$W_{in,i}^{t,init} = \frac{P_{X,i}^t + W_{in}^{t-1}}{2}, \quad W_{out,i}^{t,init} = \frac{P_{Y,i}^t + W_{out}^{t-1}}{2}$$

This computation is performed for $i = 1, \cdots, C$ and aggregated in order to obtain the updated weights that are used to evaluate the FL performances at the t^{th} federation step, assuming $W_{in}^0 = 0$ and $W_{out}^0 = 0$.

3.4.2 Sequential Updates

In the case of the sequential updates, the process is performed in a sequential manner. Three steps of the updating process are as follows:

1. The NNPLS model training process starts at a randomly selected client service, assuming that this client has sufficient data to start the training process.
2. The central service unit collects the model weights sent by that client service.
3. The next client service unit, which is ready to perform the model training process, downloads the weights from the central service unit and update its NNPLS model.

This process continues sequentially and one federating step is completed when the model of all the client services is up-to-date. The weights from the client service that is used for performing the last training is used as the final weights of the FL-NNPLS system, which will be utilized for the next federating step.

Let us assume the final NNPLS model weights from the $(t-1)^{th}$ federation step are W_{in}^{t-1} and W_{out}^{t-1}. At the t^{th} federation step, the first model training is performed by the i^{th} client, where $i \in \{1, \cdots, C\}$. The initial weights are computed as follows:

$$W_{in,i}^{t,init} = \frac{P_{X,i}^t + W_{in}^{t-1}}{2}, \quad W_{out,i}^{t,init} = \frac{P_{Y,i}^t + W_{out}^{t-1}}{2}$$

where $P_{X,i}^t$ and $P_{Y,i}^t$ are the loading matrices of the dataset $\{X_i^t, Y_i^t\}$) of the i^{th} client service at the t^{th} federation step. The NNPLS model is then trained to compute the final weights. The t^{th} federation step is completed once all client services have finished updating the model. This procedure is continued for $t = 1, \cdots, T$ assuming that at $t = 1$, $W_{in}^0 = 0$ and $W_{out}^0 = 0$.

3.5 Evaluation Metrics

The metrics, Root Mean-Square Error ($RMSE$) and coefficient of determination (R^2) are used to evaluate the predictive accuracy of the LSQR, PLSR, and NNPLS models under the federated and non-federated (i.e., CML) approaches.

The $RMSE$ quantifies the standard deviation of the residuals and computed as:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{N-1}},$$

where y and $\hat{y}$ are actual and the predicted response variable and N is the sample size.

The R^2 depicts the proportion of variance in the response variable y, which is related to the predictor variables in X. Therefore, we use R^2 as the accuracy measure to represent

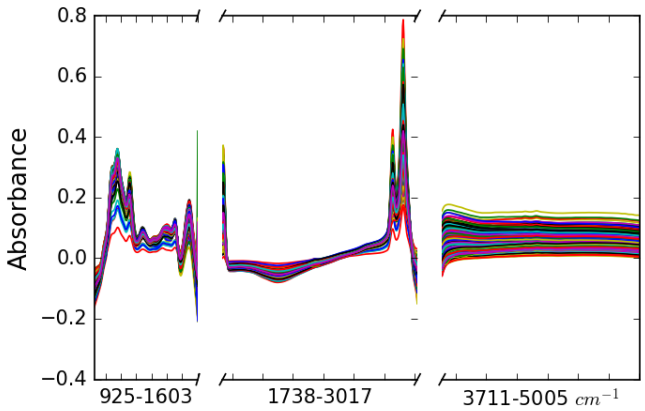


Fig. 5. Water absorbance regions removed MIR spectra of 712 milk samples within the wave number region $2500 - 25000nm(900 - 5000cm^{-1})$ of the electromagnetic spectrum.

how accurate a ML model can predict a response variable y in our evaluations and compute as:

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2},$$

where $y, \hat{y}$, N have the similar meaning as explained under $RMSE$.

4 Performance Evaluations

In this section, we explore the NNPLS model performances compared to the ML models of LSQR and PLSR and also, compare the learning performances of the NNPLS model based FL and non-FL (i.e., CML) approaches. To perform these experiments, we use a dataset of MIRS from bovin milk. Initially, the dataset is briefly introduced, including the pre-processing steps. Based on the predictive performance obtained using a LSQR model, we explain the different characteristics of FL. This is followed by a discussion on the non-federated CML performance of the NNPLS model relative to LSQR and PLSR models using three different predictive parameters. Finally, we examine the performance of the FL-NNPLS approach for the same predictive parameters.

When performing distributed learning using the dataset (i.e., training the NNPLS model under FL settings), we divid the MIRS dataset among five distributed clients. The number of clients was limited to 5 in order to have a sufficient number of samples per client (means client service) to perform the learning. This is because the original MIRS dataset consists of 712 samples only (each subset has $712/5 \approx 140$ samples). In a real-world scenario, this is feasible because in general, the herd size of an average dairy farms is around 100-150. The sub-samples are collected from client services into a central location (e.g., Cloud infrastructure) when performing CML, and this is assuming no privacy concern are raised for the communicating data. In all model training processes, 80% of the total samples were randomly selected for training and the remaining samples were used for testing.

4.1 MIRS Dataset and Pre-processing

The data used in this study originated from the Teagasc research dairy farm at Moorepark, Ireland where MIR spectra of milk were collected. The composition of milk was determined using the FOSS MilkScan [44]. The dataset consists of MIR spectra of 712 different milk samples in the wavenumber region of $925 - 5005cm^{-1}$ with a resolution of $3.853cm^{-1}$. The wavenumbers were rounded to the nearest integer. As a result, the given spectrum contained 1060 transmittance data points. Hence, the original gold standard MIRS spectra used for FL was a 712×1060 size matrix. We converted them to absorbance values by taking log_{10} from the reciprocal of the transmittance values. The absorbance values in the milk samples indicates the amount of absorption of the electromagnetic radiation when the MIR light penetrates through the milk sample. Higher absorbance values indicate that the MIR light penetrates less at certain wavenumbers according to the molecular bonds. In addition, the percentages of the selected milk nutrient components (MQTs), Lactose, Fat, and Protein, corresponding to each milk sample were stored in a matrix ($y_{n\times k}$, where $n = 712$ and $k = 3$).

In spectrometry-based data analysis, pre-treatments are necessary as MIRS data contains large quantity of redundant data which adds variability in the wavelengths. Also, the higher dimensionality and multi-collinearity among the wavelengths limits the use of simple ML models. Consequently, these factors could affect the resulting predictive accuracy. The original milk spectrum indicated two random sharp fluctuation regions, which occurs in the wave number regions of $1500 - 1800cm^{-1}$ and $2900 - 3800cm^{-1}$ per visual observation (see Fig.5). These regions are the water absorbance regions according to the pure water spectrum at $25°C$, which corresponds to $O = H$ bonds in the spectrum. We identified these two regions based on PLSR model calibration, which was conducted on our gold standard data and removed it in the pre-processing stage. Then the water absorbance regions removed were $1607 - 1734cm^{-1}$ and $3021 - 3707cm^{-1}$ [17], [18]. By removing these two regions, the dimensionality of water free spectrum (X) was 712×847. Fig. 5 represents the water absorbance regions removed spectra.

Scaling MIRS data was not a compulsory approach since all the features were in the units of absorptions. Therefore, the water-removed MIRS data was then fed into PCA dimension reduction stage and the PCs corresponding to the reconstruction error less than 10^{-4} were selected; reconstruction-error is the l_2 norm of $(X - \hat{X})$, where $\hat{X}$ is the reconstructed X by PCA. These pre-processing steps could precisely remove the wavenumbers from the original spectra to obtain pre-processed MIR spectra (say X) for use in FL. We used Python scikit-learn and tensor flow libraries for all the analytical work.

4.2 FL performance of MIRS Data with a LSQR model

A LSQR model was formed by including all the parameters in X (i.e., PCs with reconstruction error $\leq 10^{-4}$) and then federated once by equally distributing the 712 samples among the 5 clients ($\approx$140 samples per client). With a 10^{-3} learning rate, the SGD algorithm was then used to train the model at each client for 10^3 iterations. The predictive performance of the FL (i.e., DML) approach (parallel) was compared with the non-FL (CML) approach. Fig. 6 shows

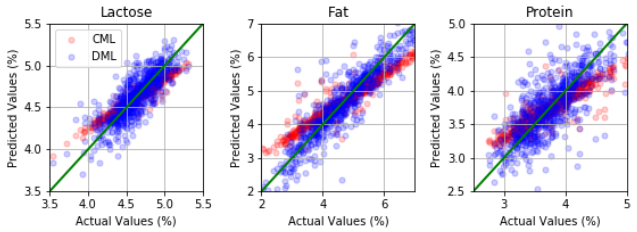


Fig. 6. Federated and centralized prediction accuracy of MQTs in the MIRS dataset (DML with 5 clients).

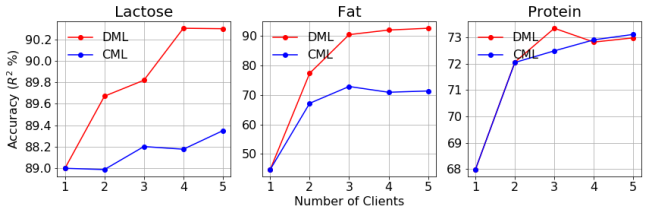


Fig. 7. Comparison of FL and CML performances of training efficiency with number of clients obtained using MQTs.

the training accuracy of CML and DML approaches for three MQTs. The predictive and actual MQT values under the DML approach are distributed around the straight line (i.e., actual y and predicted y ($\hat{y}$) should be linearly correlated as $y = m\hat{y}$, where $m = 1$) compared to those under the CML approach. That is, the DML (FL) predictive performance is better than that of the CML with LSQR model.

Following the similar procedure used above, the variability in predictive performance of FL with increasing number of clients was studied with the results summarized in Fig. 7. The FL approach achieved higher accuracy and converged faster than the CML approach. Therefore, having many clients contributes to improving the FL efficiency by speeding up the model training process. The effectiveness of the FL approach mainly depends on the number of clients as well as the number of samples used by each client to train the ML model. Having a set of suitable initial model parameter values generally guarantees better accuracy when compared to their optimal values. So the updated set of model parameters that are available at the central processing unit (e.x., server) is sent to the next client(s) as initial parameter values. The convergence of SGD gradually becomes faster as federation progresses.

These results confirm that FL performs better than the CML approach, but the performance can be improved further with increasing number of clients as well as using larger training samples. However, the limitations of LSQR mentioned in Section 3.1 might cause a poor predictive performances for some MQTs [17]. The next section explains how to overcome those limitations based on the computational evidence derived from using PLSR and NNPLS models.

4.3 Centralized (non-FL) and FL performances with NNPLS model

In this section, non-federated (CML) performance of the NNPLS model is compared to the LSQR and PLSR models. Then the performance of FL-NNPLS is explored.

Table 1 represents the non-FL predictive accuracy obtained from the LSQR, PLSR, and NNPLS models for each

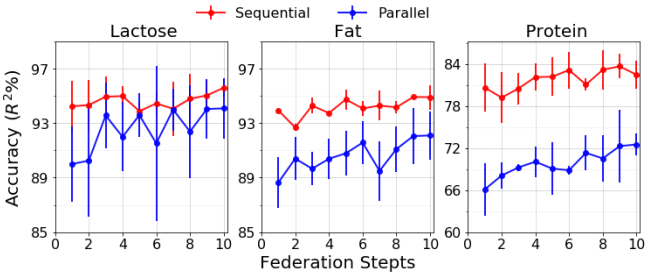


Fig. 8. NNPLS model-based FL performance for different MQTs under the sequential and parallel model updating approaches.

MQT, including the number of LVs. The CNN model is explained in the next section. The replacement of the LSQR model by the PLSR and then the NNPLS models contributed to improving the predictive accuracy of all MQTs; the largest improvement was for Protein, followed by Lactose. This is highly likely attributed to the PLSR and NNPLS models taking into account the multicollinearity in the MIRS data which is not considered in the LSQR model. Also, the NNPLS model is capable of capturing non-linear functional relationships in addition to multicollinearity in MIRS data. Moreover, NNPSL model follows the SGD optimization to select the optimal parameter values, which is not considered in the PLSR model. That is why the accuracy of each MQT obtained from the NNPLS model is better to that of the LSQR and PLSR models. Thus, the NNPLS model is computationally more effective compared to the traditional NN models and also able to provide more precise learning outcomes compared to LSQR and PLSR.

With five clients, the NNPLS model of each MQT was federated for ten times under the sequential and parallel updating approaches separately, assigning 140 ($\approx 712/5$) samples randomly for every client at each federation. This approach is used in the same training and validation settings, which were used under the non-FL approach. Fig. 8 depicts the variability in the average predictive accuracy over the five clients obtained from both updating procedures at each federation step. In general, with increasing number of federation steps, the predictive accuracy of all MQTs increased in both updating approaches. The accuracy obtained from the sequential updating was, however, higher than the parallel updating approach. Furthermore, after ten federation steps, all MQTs achieved higher accuracy compared to the non-FL accuracy given in Table 1 for both updating approaches.

In general, the NNPLS model performed better than the LSQR and PLSR models with MIRS milk data. At the same time, the FL-NNPLS approach achieved greater performance compared to the performance obtained from the CML approach. Moreover, the sequential updating based FL performance has better performance compared to the parallel updating based FL approach.

5 Discussion

The FL method can be considered as a realization of the concept of *Data Gravity* proposed by *Dave McCrory* in 2010. He pointed out that with increasing data sizes, the computational power should be shifted towards the data sources. The Fog/edge computing and Cloudlets are

MQT	LSQR (%)		LVs	PLSR (%)		NNPLS (%)		CNN(%)	
	Train:	Vali:		Train:	Vali:	Train:	Vali:	Train:	Vali:
Lactose	92.43	91.62	12	94.05	90.86	96.41	93.39	97.85	92.58
Fat	93.49	87.50	5	93.95	91.44	96.11	91.12	96.36	92.21
Protein	83.86	66.87	5	75.62	69.44	87.46	83.09	82.05	77.66

TABLE 1
Centralized (non-FL) training (Train:) and validation (Vali:) accuracy ($R^2\%$) of LSQR, PLSR, NNPLS, and CNN models for different milk quality parameters (MQTs).

two popular DML enablers where this concept is being practiced [18], [45]. In these DML techniques, FL is one of the latest optimization approaches which can be used to perform analytics. Therefore, in this section, first we discuss the state-of-the-art performances of the FL-NNPLS approach by using a deep learning technique, which is the Convolutional NN (CNN). The common issues of the DML approach are then discussed, including some of the challenges in FL-NNPLS technique. Subsequently, one of the critical challenges, which is data imbalance is discussed. Finally, directions for future research under FL are briefly discussed.

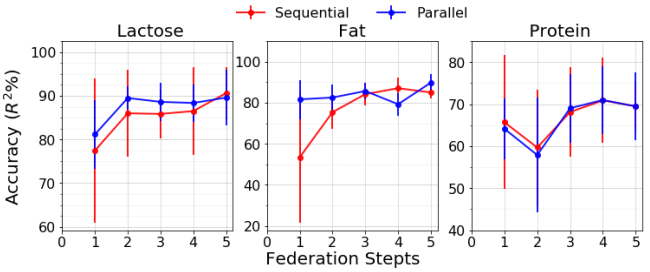


Fig. 9. CNN model-based FL performances under the sequential and parallel updating techniques.

5.1 Comparison of FL-NNPLS model performance to a deep NN model

Considering the different CNN models (e.g., LNet, Vgg-19, and Resnet), which are explained in [18], [46], [47], the CNN model was selected based on the LNet-5 and Vgg-19 models such that they are deeper (i.e., number of layers) than the LNet-5, but not as deeper as Vgg-19. This means that the CNN model consisted of three convolution and three dense layers. Each convolution layer contained a 3×3 sized kernel and the number of features extracted from each layer was 20, 30, and 40, respectively. Also, each convolution layer was followed by a max-pooling layer with a kernel of size 2×2. A flatten layer was included after the last convolution layer, and then the dropout layer dropped out by 20% neurons of the flatten layer. The first dense layer contained 30 neurons. The number of neurons in the second dense layer was equal to the number of LVs of the predictor variable used for learning. The last dense layer contained only one neuron and used the *linear* activation function; nevertheless all other layers of the CNN model used the *rectified* activation function. The *ADAM* optimization technique was used to train the CNN model. The reason behind selecting a CNN model architecture between LNet-5 and Vgg-19 was that training large deep learning models like Vgg-19 under the FL settings may not be feasible under certain circumstances, such as limited resources and low complexity.

Under the CML settings, the water-free MIRS dataset was first compressed by applying PCA with 10^{-4} reconstruction error and then the compressed dataset fed into the CNN model. The model was trained for 10^3 times by selecting the initial network weights from the uniform distribution. The predictive accuracy for each MQT was then computed and given in Table 1. It is clear that the NNPLS model has state-of-the-art performance because the CNN model outcomes are comparable to the NNPLS model.

To train the CNN model under the FL settings, the same procedure which was used to train the NNPLS model in Section 4.3 was followed. However, the initial network weights were selected from the uniform distribution. If the square root of the selected number of PCs was not an integer, then the PCA compressed data could not reshape it to feed to the CNN model for training and validation. Therefore, the zero padding technique was used to adjust the number of feature variables in the compressed dataset before reshaping. The number of samples per client was increased up to 250, but the federation steps were limited up to five as over-fitting is a common issue with the CNN model due to the small data size. Fig. 9 represents the validation accuracy for each MQT with increasing number of federation steps under the sequential and parallel updating approaches. As shown in the results, the accuracy improved for all MQTs under two updating approaches.

Comparing the CNN-based FL performance to the FL-NNPLS outcomes, the NNPLS model achieved higher accuracy for all MQTs compared to the CNN model under similar FL settings. Moreover, the convergence efficiency of the NNPLS was faster than the CNN model, as CNN requires longer training time to achieve similar performance to the FL-NNPLS. Therefore, these results prove that the NNPLS model results in greater performance compared to the CNN model under similar experimental settings. Fig. 10 depicts the sequential and parallel updating performances based on the validation accuracy of each MQT obtained from the NNPLS and CNN models. While the variability in the predictive accuracy of all the MQTs at each client is displayed in Fig. 10(a), Fig. 10(b) represents the change in predictive accuracy for each MQT with each federation steps. In general, the predictive performance from the sequential updating method was higher than the parallel updating method for both models. Also, under the sequential updating technique, the NNPLS model based predictive accuracy was comparable to that from the CNN model for all MQTs. However, considerable differences were observed for certain MQTs with the parallel updating approach. Therefore, it can be concluded that FL with the sequential updating performs well compared to the parallel updating. Also, the NNPLS model has comparable performance to the state-of-the-art

CNN-based deep learning model based on the predictive performance obtained for the MIRS data of milk.

5.2 Advantages of FL-NNPLS

The DML framework based on the FL-NNPLS approach can handle most of the issues mentioned heretofore. Since data will not migrate from the data sources, and ML model updates do not store them in the server, the FL-NNPLS optimizes resource consumption as well as data privacy, security, and data ownership. Also, as every client integrates updated ML model immediately after each federation, they can make timely decisions effectively. In many applications, NNs are the state-of-the-art though selecting a proper network configuration is time and resource-consuming. However, modern deep learning models ResNnet and Googlenet can be easily optimized and gain higher accuracy with increasing depth of the models and can control the computational cost required with deep learning models, respectively [46]. Employing them in Fog/Edge computing based DML systems might not be possible particularly in smart farming sectors, because such systems mostly rely on resource limited sensor nodes which cannot train those models. However, considering the applicability of modern DML frameworks such as *Horovod* [1] will be a good alternative to handle such limitations.

Since NN used in the FL-NNPLS approach is associated with the PLSR technique in selecting suitable NN configurations, the learning process is faster and preserves the computational power. Having a proper set of initial network weights speeds up the network convergence, cutting down the computational burden required for deciding the NN configurations, providing a faster convergence rate. Since the PLSR technique provides these pre-requirements, NNPLS-based ML can provide computationally inexpensive, robust, and scalable solutions for a broader range of applications. Furthermore, PLSR effectively overcomes the multicollinearity and higher dimensionality, while NN enables capturing complex functional relationships in the data. Therefore, the scalability of the FL-NNPLS approach is better compared to the NN or PLSR methods.

5.3 General Limitations in FL-NNPLS Approach

The ultimate purpose of this DML is for deriving meaningful and timely insights from massively distributed datasets. Hence the necessity for efficient FL frameworks has a growing demand with the growing prevalence of big data in a broad range of applications. Nevertheless, there are some common critical issues associated with it, and some are listed as follows:

1. The FL framework is used for learning from large-scale distributed data, but finding resources in order to respond to the ever-increasing data volume is challenging.
2. The datasets involved in FL are typically heterogeneous and that brings up constraints such as aggregating the FL model parameters and defining a common representation for data to be able to apply ML. Also the datasets are not complete, balanced, and uncertain due to a number of reasons such as missing data or their unavailability, and the un-verifiability of all data sources. As a result, most ML algorithms cannot be applied directly so that deriving precise insights from such data could be challenging.
3. The FL system is totally dependent on the coordination device which provides services to the collection by the local model updates and using this to produce the global model updates. Any functional failure (or misbehaviour) of this entity could result in a collapse of the entire system.
4. It has been already warned that significant information can be extracted by tampering the model updates [48]. Hence, extra security efforts are now essential, particularly in smart farming applications for performing safe communication of model updates between the clients and the central service units.
5. The FL system is lacking a proper mechanism to examine the validity of the clients' data and model updates as they can inject false information into the FL system.

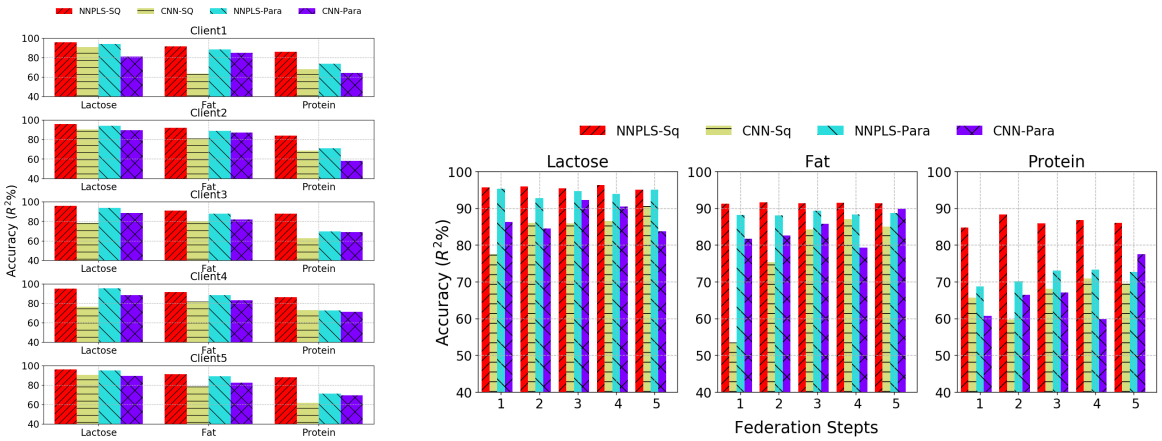
Therefore, FL systems should have the potential to understand these factors and be equipped with the necessary tools in order to efficiently overcome them. For example, to overcome the data imbalance issue in FL, we propose an approach by using the FL-NNPLS method and explore its performance based on the MIRS dataset.

5.3.1 Data imbalance issue in FL-NNPLS

Data imbalance is commonly encountered in many ML applications most notably in data classification and DML. This happens when the number of clients that have small quantity of samples is significantly higher compared to the total number of clients. Consequently, ML outcomes are most likely biased towards the clients which have larger samples. ML algorithms such as data classification consider the clients are having a small number of samples (minority samples/classes) as noises and tends to neglect them during the learning process. In fact, minority classes are usually more critical. For instance, in dairy herds, there could be very few sick cows relative to the number of healthy cows, and this minority group plays a crucial role when identifying animals which are affected. Hence, data imbalance is an important topic in advanced ML research with the growing interest in DML. According to a recent study [49], there are two main approaches to overcome this issue; data-level and algorithm-level approaches. While the first approach overcomes data imbalance by using re-sampling techniques, different techniques such as adding penalty constants are used in the second approach to overcome the data imbalance issue. Many experimental attempts have proved that re-sampling, i.e., over-, under-, and hybrid-sampling, is a promising way to manage data imbalance [49]. However, the best-suited technique depends on the characteristics of the imbalanced datasets.

On the other hand, there is another problem associated with FL-NNPLS approach due to data imbalance; the number of LVs varies with the data size, and consequently the NNPLS model configurations vary over the clients and it

1. https://github.com/horovod/horovod



(a) Predictive accuracy of every MQT at each client.

(b) Variation of predictive accuracy with federation steps.

Fig. 10. Sequential and parallel updating based FL performances obtained from NNPLS and CNN models for three different MQTs.

hampers aggregation of model updates. That is, feature variables are heterogeneously distributed over clients and DML system aggregates different ML models in order to perform learning. In some studies, DML with different ML models has been named as vertically partitioned data or dimensionally distributed data [50]. However, these ML models that include the re-sampling methods cannot solve the imbalanced data issue alone in FL-NNPLS approach. Hence, a technique, which can control both issues at the same time, is required. Therefore, in the present study, a re-sampling and zero-padding based joint approach was used to overcome the issue.

Five clients were federated for five times under the sequential updating FL approach since it performed well compared to the parallel approach. In each federation step, the number of samples of every client was allowed to vary randomly between 50-250. PCA reconstruction error was set to 10^{-4} in order to select the optimal number of PCs. The same procedure was used in Section 3.4.2, but two additional steps were used here. The first one is to balance sample sizes of the clients. The random re-sampling was applied to the current FL participator only if its sample size was less than the previous client which performed model updating. Randomly selected samples from its own samples were used to balance the current sample size to the sample of the previous client (i.e., random up-sampling). The second step was added to the NN weight aggregation stage in which zero-padding was applied to equalize the sizes of the NN weight matrices. The variability in predictive accuracy was computed under both the CML and FL approaches.

Fig. 11 represents the federated and non-federated predictive accuracy obtained for the milk Fat at each federation step for every client, including the original sample sizes and LVs. The predictive accuracy from the FL approach is generally higher than the non-FL. Therefore, it seems that the re-sampling and zero padding based approach has the potential of mitigating data imbalance issues. However, further research is essential to study the validity of this approach in different applications.

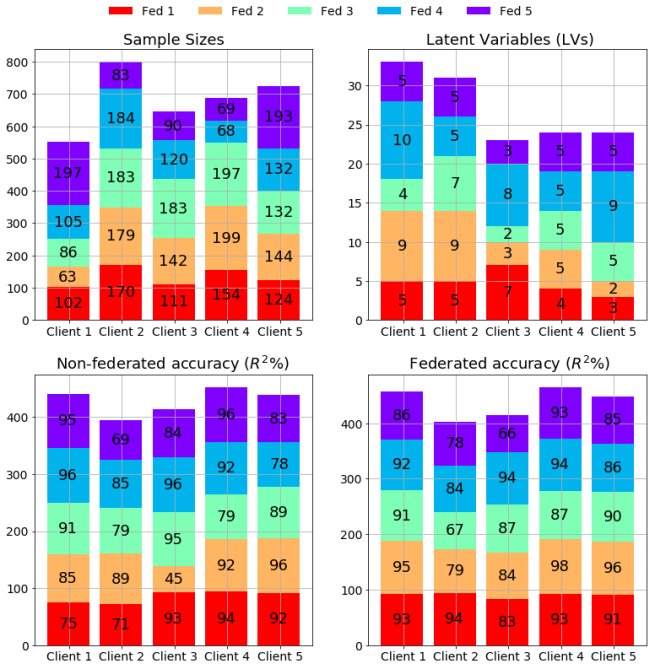


Fig. 11. The federated and non-federated NNPLS model performances with imbalance data.

5.4 Future Research Directions

The future research directions can be considered in two ways; 1) finding solutions to overcome the challenges mentioned in Section 5.3 and 2) exploring novel approaches to enhance the performances of FL-NNPLS in distributed services.

Some of the possible solutions could be explored further to overcoming these challenges are:

1. Exploring novel cooperative computing (resource sharing) approaches such as offloading computations to neighbouring devices with services as explained in [51] for minimizing the constraints in resources. Also, research in effective data compression techniques for compressing training data as well as communication over the FL system would be an interesting approach for minimising the computing and com-

munication costs. In addition, incorporating the distributed cloud service model for resource allocation proposed in [52] and mobile edge computing approach presented in [45] would be good approaches to minimize resource limitations.

2. Designing techniques for FL with dynamically changing central service unit based on factors such as resource availability and communicability with the client's services would also contribute to improving the stability of the FL system, resulting in the minimization of the impact of failure (or misbehaviour) of the central service unit. The concepts about optimizing the node failure in connected devices given in [53] would be beneficial exploring a path for mitigating the impact of central service unit failure in FL.
3. In order to prevent tampering the model updates of the services traversing over the FL system, public-private key based data encryption techniques such as the authenticated symmetric encryption with Diffie-Hellman (D-H) key exchange service can be incorporated with FL. Further exploration on how they facilitate to improving data privacy and communication resources will be required.
4. Proposing techniques for integrating FL with blockchain services, for validating model updates and also minimizing the misbehaviour of any FL participant.
5. Performing FL by selecting only a set of clients' services, who have good history of providing valid model updates, to update the final ML model will also be a promising solution to control the injection of false information to the FL system.

There are different ways for developing novel approaches to enhance the FL performance in distributed services.

1. One of the main goals of the FL approach is to achieve high accuracy with a minimum number of federated communication rounds. This goal can be achieved by using the ML algorithms which have faster convergence rates. Hence, developing a better theoretical understanding about the convergence properties of such algorithms will be an interesting future research direction.
2. Since FL is an optimization framework, a convex optimization function is essential to guarantee the convergence of the model parameters. However, NN-based ML yields non-convex function so there is no evidence regarding the guarantee of convergence of the optimization algorithm. Therefore, studies of the FL problems for non-convex objectives is another direction that may contribute to the efficient application of FL for solving even more complex problems in advanced applications such as object detection and image processing.
3. Developing a business model based on a token-based FL approach. This means that the clients who contributes to update the ML model can rent the model for outsiders who want the model for practicing different services such as making decisions. In return, they have to pay a certain amount of tokens, through a form of currency, to the FL client(s). This method will facilitate FL-based platforms to provide greater services to wider range of application domains.
4. The present study assessed the predictive performance of the NNPLS model based CML and DML only. However, exploring the computational efficiency would be an interesting extension of this work by taking into account various factors. Some of the factors, which could contribute towards improving the efficiency of the FL-NNPLS approach, are resource availability at the client services and central service unit, frequency of data collection, and model update transmission cost.

6 Conclusion

This paper presents the applicability and benefits of using a hyrid model of FL and ML models for distributed services. A particular application of the services is a case study on spectral data generated from milk samples, which essentially operates as a tool to predict three milk quality parameters. The NNPLS model developed for the FL model based on the limitations of the LSQR, PLSR, and NN models was used for predictions under the CML and DML settings. Under the CML settings, the NNPLS model contributed to the improvement of the predictive performance compared to the LSQR and PLSR models and also achieved comparable performances to the state-of-the-art CNN model. Therefore, the NNPLS model is a good fit for performing predictive analytics on milk quality data. Under the FL configurations, our NNPLS model achieved similar performance when compared to the CML approach, and by only using a few federation steps. Moreover, with the similar FL settings, FL performance of the NNPLS model was similar to the state-of-the-art CNN model. Therefore, FL-based NNPLS model can provide timely insights regarding the composition of milk while preserving data privacy and ownership with minimal resource requirements which are critical challenges in providing effective services in modern day smart farming applications. The sequential updating based FL approach is a good fit for analyzing milk composition as it achieves better performance with both the NNPSL and CNN models compared to the parallel updating approach. The resampling and zero-padding based approach contributed to mitigating the impact of data imbalance in FL. However, further investigation is required to improve the performance further.

Acknowledgment

This research was supported by the Science Foundation Ireland (SFI) project "PrecisionDairy (ID: 13/1A/1977)" as well as a research grant from Science Foundation Ireland and the Department of Agriculture, Food and Marine on behalf of the Government of Ireland under the Grant 16/RC/3835 (VistaMilk).

References

[1] C. Kulatunga, L. Shalloo, W. Donnelly, E. Robson, and S. Ivanov, "Opportunistic wireless networking for smart dairy farming," *IEEE IT Professional Magazine*, vol. 19, no. 2, 2017.

[2] S. Capalbo, J. Antle, and C. Seavert, "Next generation data systems and knowledge products to support agricultural producers and science-based policy decision making," *Elsevier J. of Agricultural Systems*, vol. 155, pp. 191–199, 2016.

[3] S. Wang, L. Huang, Y. Nie, X. Zhang, P. Wang, H. Xu, and W. Yang, "Local differential private data aggregation for discrete distribution estimation," *IEEE Transactions on Parallel and Distributed Systems*, pp. 1–1, 2019.

[4] H. O. Alanazi, A. H. Abdullah, and K. N. Qureshi, "A critical review for developing accurate and dynamic predictive models using machine learning methods in medicine and health care," *Medical,Systems*, vol. 14, no. 4, 2017.

[5] "Department of Agriculture and Food and the Marine and Ireland, Food Wise 2025 - Local Roots Global Reach: A vision for growth for the Irish agricultural economy for the next 10 years," https://www.agriculture.gov.ie/foodwise2025/, 2018.

[6] J. Qiu, Q. Wu, G. Ding, Yuhua, and S. Feng, "A survey of machine learning for big data processing," *EURASIP J. of Advances in Signal Processing*, 2016.

[7] P. Sun, Y. Wen, T. N. B. Duong, and S. Yan, "Timed dataflow: Reducing communication overhead for distributed machine learning systems," in *IEEE Conf. on Parallel and Distributed Systems*, 2016.

[8] F. Jalali, K. Hinton, R. Ayre, T. Alpcan, and R. S. Tucker, "Fog computing may help to save energy in cloud computing," *IEEE J. of Selected Areas in Communications*, vol. 34, no. 5, 2016.

[9] E. P. Xing, Q. Ho, P. Xie, and D. Wei, "Strategies and principles of distributed machine learning on big data," *J. of Engineering*, vol. 2, pp. 179–195, 2016.

[10] J. Konecny, H. B. McMahan, D. Ramage, and P. Richtarik, "Federated Optimization: Distributed Machine Learning for On-Device Intelligence," https://arxiv.org/abs/1610.02527, 2016.

[11] J. Lu, S. Chen, W. Wang, and H. v. Zuylen, "A hybrid model of partial least squares and neural network for traffic incident detection," *Elsevier J. of Expert Systems with Applications*, vol. 39, pp. 4775–4784, 2012.

[12] A. Debiolles, L. Oukhellou, and P. Akrin, "Combined use of partial least square regression and neural network for diagnosis tasks," in *IEEE Conf. on Pattern Recognition (ICPR'04)*, 2004, pp. 1051–4651.

[13] G. Visentin, A. McDermott, S. McParland, D. P. Berry, O. A. Kenny, A. Brodkorb, M. A. Fenelon, and M. D. Marchi, "Prediction of bovine milk technological traits from mid-infrared spectroscopy analysis in dairy cows," *Elsevier J. of Dairy Science*, vol. 98, 2015.

[14] M. Taghdiri, G. Karim, S. Safi, A. R. Foroushani, and A. Motalebi, "Study on the accuracy of milk amyloid a test and other diagnostic methods for identification of milk quality," *Vet Research Forum*, vol. 9, no. 2, pp. 179–185, 2018.

[15] S. McParland and D. P. Berry, "The potential of fourier transform infrared spectroscopy of milk samples to predict energy intake and efficiency in dairy cows," *Elsevier J. of Dairy Science*, vol. 99, 2016.

[16] L. Hedlund and H. Lovlie, "Personality and production: Nervous cows produce less milk," *Elsevier J. of Dairy Science*, vol. 98, no. 9, pp. 5819–5828, 2015.

[17] D. Vimalajeewa, D. P. Berry, E. Robson, and C. Kulatunga, "Evaluation of non-linearity in mir spectroscopic data for compressed learning," in *IEEE ICDM Workshop on High Dimensional Data Mining (ICDM-HDM)*, 2017.

[18] D. Vimalajeewa, C. Kulatunga, and D. P. Berry, "Learning in the compressed data domain: Application to milk quality prediction," *Elsevier J. of Information Sciences*, vol. 459, pp. 149–167, 2018.

[19] J. Kang, R. Yu, X. Huang, S. Maharjan, Y. Zhang, and E. Hossain, "Enabling localized peer-to-peer electricity trading among plug-in hybrid electric vehicles using consortium blockchains," *IEEE Trans. on Industrial Informatics*, vol. 13, no. 6, pp. 3154–3164, 2017.

[20] T. S. Brisimia, R. Chena, T. Melac, A. Olshevskya, I. C. Paschalidisa, , and W. Shi, "Federated learning of predictive models from federated electronic health records," *J. of Medical Informatics*, vol. 112, pp. 57–69, 2018.

[21] S. Wolfert, L. C. Verdouw, and M. Bogaardt, "Big data in smart farming: A review," *Elsevier J. on Agricultural Systems*, vol. 153, pp. 60–80, 2017.

[22] Y. Jiber, H. Harroud, and A. Karmouch, "Precision agriculture monitoring framework based on wsn," in *2011 7th International Wireless Communications and Mobile Computing Conference*, 2011, pp. 2015–2020.

[23] V. Sharma, I. You, F. Palmieri, D. N. K. Jayakody, and J. Li, "Secure and energy-efficient handover in fog networks using blockchain-based dmm," *IEEE Communications Magazine*, vol. 56, no. 5, pp. 22–30, 2018.

[24] H. Zeng, S. R. Kulkarni, and H. V. Poor, "Attribute-distributed learning: Models, limits, and algorithms," *IEEE Tran. on Signal Processing*, vol. 59, pp. 386–398, 2011.

[25] M. Li, D. G. Andersen, J. W. Park, A. J. Smola, A. Ahmed, V. Josifovski, J. Long, E. J. Shekita, and B. Su, "Scaling distributed machine learning with the parameter server," in *USENIX Symposium on Operating Systems Design and Implementation*, 2014.

[26] C. Xu, K. Wang, P. Li, S. Guo, J. Luo, B. Ye, and M. Guo, "Making big data open in edges: A resource-efficient blockchain-based approach," *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 4, pp. 870–882, 2019.

[27] J. Dean and S. Ghemawat, "Mapreduce: Simplified data processing on large clusters," *J. Communications of the ACM*, vol. 51, pp. 107–113, 2008.

[28] " Apache Hadoop," http://hadoop.apache.org/, 2018.

[29] J. Konecny, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," in *IEEE Neural Information Processing Systems Workshop*, 2016.

[30] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," *arXiv:1602.05629*, vol. 2, 2016.

[31] G. Lan, S. Lee, and Y. Zhou, "Communication-Efficient Algorithms for Decentralized and Stochastic Optimization," https://arxiv.org/abs/1701.03961, 2017.

[32] R. Hasegawa and K. Hotta, "Plsnet: Hierarchical feature extraction using partial least square regression for image classification," *IEEE Tran. on Electrical and Electronic Engineering*, vol. 12, pp. 91–96, 2017.

[33] T. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, and Y. Ma, "Pcanet: A simple deep learning baseline for image classification?" *IEEE Tran. on Image processing*, vol. 24, pp. 5017–5032, 2014.

[34] R. Hasegawa and K. Hotta, "Plsnet: A simple network using partial least square regression for image classification," in *IEEE Conf. on Pattern Recognition (ICPR)*, Cancun Center, Cancun, Mexico, 2016.

[35] M. Gowanlock, C. M. Rude, D. M. Blair, J. D. Li, and V. Pankratius, "A hybrid approach for optimizing parallel clustering throughput using the gpu," *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 4, pp. 766–777, April 2019.

[36] K. Song, L. Li, S. Li, L. Tedesco, H. Duan, Z. Li, K. Shi, J. Du, Y. Zhao, and T. Shao, "Using partial least squares artificial neural network for inversion of inland water chlorophyll-a," *IEEE Tran. on Geoscience and Remote Sensing*, vol. 52, 2014.

[37] R. Hasegawa and K. Hotta, "Stacked partial least square regression for image classification," in *3rd Asian Conf. on Pattern Recognition*, 2015.

[38] V. D. Justafort, R. Beaubrun, and S. Pierre, "A hybrid approach for optimizing carbon footprint in intercloud environment," *IEEE Transactions on Services Computing*, vol. 12, no. 2, pp. 186–198, 2019.

[39] O. B. Akan, O. Cetinkaya, C. Koca, and M. Ozger, "Internet of hybrid energy harvesting things," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 736–746, 2018.

[40] D. C. Rose, W. J. Sutherland, C. Parker, M. Lobley, M. Winter, C. Morris, S. Twining, C. Ffoulkes, T. Amano, and L. V. Dicks, "Decision support tools for agriculture: Towards effective design and delivery," *Elsevier J. of Agricultural Systems*, vol. 149, pp. 165–174, 2016.

[41] N. Alexandratos and M. Bruinsma, "World agriculture towards 2030/2050, food and agriculture organization in the united nations," in *EAS working Paper No 12-03*, 2012.

[42] D. Vasisht, Z. Kapetanovic, J. Won, X. Jin, R. Chandra, A. Kapoor, S. N. Sinha, M. Sudarshan, and S. Stratman, "Farmbeats: An iot platform for data-driven agriculture," in *USENIX Symposium on Operating Systems Design and Implementation*, 2017.

[43] P. H. Garthwaite, "An interpretation of partial least squares," *American Statistical Association*, vol. 89, no. 425, pp. 122–127, 1994.

[44] H. Winning, "Standardization of ftir instruments using foss," *FOSS*, vol. 1, 2004.

[45] Q. Wang, S. Guo, J. Liu, C. Pan, and L. Yang, "Profit maximization incentive mechanism for resource providers in mobile edge computing," *IEEE Transactions on Services Computing*, pp. 1–1, 2019.

[46] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[47] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.
[48] J. Weng, J. Weng, M. Li, Y. Zhang, and W. Luo, "Deepchain: Auditable and privacy-preserving deep learning with blockchain-based incentive," *IACR Cryptology ePrint Archive*, vol. 2018, p. 679, 2018.
[49] G. Y. Wong, F. H. F. Leung, and S.-H. Ling, "A hybrid evolutionary pre-processing method for imbalance datasets," *Elsevier J. of Information Sciences*, vol. 454, pp. 161–177, 2018.
[50] H. Zheng, S. R. Kulkarni, and H. V. Poor, "Dimensionally distributed learning models and algorithm," in *IEEE Conf. on Information Fusion*, Cologne, Germany, 2008.
[51] C. Kulatunga, K. Bhargava, D. Vimalajeewa, and S. Ivanov, "Co-operative in-network computation in energy harvesting device clouds," *Sustainable Computing: Informatics and Systems*, vol. 16, pp. 106–116, 2017.
[52] K. Metwally, A. Jarray, and A. Karmouch, "A distributed auction-based framework for scalable iaas provisioning in geo-data centers," *IEEE Transactions on Cloud Computing*, pp. 1–1, 2018.
[53] M. M. Tajiki, M. Shojafar, B. Akbari, S. Salsano, M. Conti, and M. Singhal, "Joint failure recovery, fault prevention, and energy-efficient resource management for real-time sfc in fog-supported sdn," *Elsevier Computer Networks*, 2019.

Dixon Vimalajeewa recieved his B.Sc in mathematics and statistics from The University of Ruhuna, Sri Lanka in 2012, and M.Sc in Computational Engineering from The Lappeenranta University of Technology, Finland in 2015.

Currently, he is a PhD student at Telecommunications Software and Systems Group (TSSG) at Waterford Institute of Technology (WIT). His research interests include data analytics, sensor-based animal phenotypes and distributed learning algorithms (dvimalajeewa@tssg.org)

Chamil Kulatunga Chamil Kulatunga received his Bachelor in Electronics and Telecommunication Engineering from the University of Moratuwa (Sri Lanka) in 1999, Masters in Computer Science from the Waterford Institute of Technology (Ireland) in 2003, and PhD in Internet Engineering from the University of Aberdeen (UK) in 2009.

He was an Experienced Postdoctoral Researcher (2015-2017) under the Science Foundation Ireland funded PrecisonDairy project at the Telecommunication Software and Systems Group, Waterford Institute of Technology, Ireland. He is currently a Research Data Analyst under the Science Foundation Ireland and Origin Enterprises funded CONSUS project at the University College Dublin, Ireland.

His current research interests include distributed machine learning and data-driven agriculture in crop and dairy farming (chamilkul@gmail.com).

Donagh P Berry received his Bachelor in Agricultural Science and PhD in quantitative genetics at Unversity College Dublin, Ireland in 2000 and 2003, respectively and a Masters in Bioinformatics and Systems Biology from University College Cork in 2012.

He is currently a senior principal investigator in quantitative geneticist at Teagasc, Ireland as well as being director of the VistaMilk Agri-Tech Research Centre. He holds professorships at three (inter)national universities. In his Teagasc capacity, he is responsible for the research on genetics in dairy cattle and is responsible for the development and implementation of genomic evaluations in dairy cattle, beef cattle and sheep in Ireland. As director of VistaMilk, he leads a team of $>$200 scientists in the development and deployment of digital technologies in precision dairy production (donagh.berry@teagasc.ie).

Sasitharan Balasubramaniam received the bachelor's degree in electrical and electronic engineering from The University of Queensland in 1998, the master's degree in computer and communication engineering from the Queensland University of Technology in 1999, and the Ph.D. degree from The University of Queensland in 2005.

He is currently an Academy of Finland Research Fellow at the Department of Electronic and Communication Engineering, Tampere University of Technology, Finland, and an Acting Director of Research at the Telecommunication Software and Systems Group, Waterford Institute of Technology, Ireland, where he was involved in a number of Science Foundation Ireland projects. His current research interests include molecular and nanocommunications, and Internet of (bio–nano) Things. He is on the Steering Committee of the ACM NanoCom Conference which he co-founded. In 2018, he received the ACM/IEEE NanoCom Outstanding Milestone Award, and he is also the IEEE Nanotechnology Council Distinguished Lecturer. He is currently an Editor of the IEEE INTERNET OF THINGS JOURNAL, Nano Communication Networks (Elsevier), and Digital Communication Networks.

Appendix E

Blockchain-Powered IoT system integrated with IoNT for Smart Farming

Journal Title:	IEEE Internet of Things
Article Type	Regular Paper
Complete Author List	Dixon Vimalajeewa, Donagh P. Berry, Subhasis Thakur, and Sasitharan Balasubramaniam
Status	Under Review

Incorporating Block Chain into Internet of Nano Things for Smart Farming

Dixon Vimalajeewa, Subhasis Thakur, John Breslin, Donagh P. Berry, Sasitharan Balasubramaniam

Abstract—

The integration of Internet of Things (IoT)-based decision-making systems with Block Chain (BC) technology can contribute immensely to autonomous Edge systems that require improved data security, auditability, and transparency. The recent introduction of the Internet of Nano Things (IoNT) can further exploit BC to improve reliability as they are able to sense molecules a more granular scale, which in turn provides a new spectrum of data that can be analyzed. The integration of IoNT into BC-powered IoT systems in any application has not yet been investigated. This study proposes a BC-powered IoNT (BC-IoNT) system for detecting chemicals at the molecular level in the context of farm management. This is a critical application for smart farming, which aims to increase sustainable farm productivity coupled with controlled delivery of chemicals. BC-IoNT system includes an analytical model formed by using the Langmuir molecular binding model integrated with the Bayesian theory. This model is used as a smart contract for detecting the level of the chemicals. Moreover, a farm credit model is used to quantify the traceability and credibility of farms to determine if they are compliant with the chemical standards. The BC-IoNT was validated through simulations to evaluate the accuracy of detecting the level of chemicals of the distributed BC approach, and it was compared to a centralized analytic approach. The accuracy of the BC-IoNT was$\geq 90\%$ and the centralized approach was $\leq 80\%$. Also, the efficiency of detecting the level of chemicals depends on the sampling frequency and variability in chemical level among farms.

Index Terms—IoT-powered Block Chain, Smart Farming, Bayesian Updating, Affinity-based Nano-Sensor, Food Supply Chain.

1 INTRODUCTION

With the progressive developments in modern Information and Communication Technologies (ICT), a wide range of application domains such as health-care, transport, and agri-tech has become further advanced resulting in novel and intelligent services. The recent advances in the Internet of Things (IoT), coupled with modern ICT, facilitates continuous monitoring of spatial and temporal variability that exist from sensors integrated into each of these application domains. In particular, sensor technologies have undergone dramatic advancements due largely to the developments in other supporting fields such as nanotechnology, which enables monitoring of molecules at a fine granular scale. The emerging field of nano-communications allows communication and networking between devices to be developed from nano-scale components [1]. The integration of nano-communications with IoT has led to a new paradigm known as the Internet of Nano-Things (IoNT), empowering the potential of creating a broader spectrum of data that can take the applications described above to unprecedented levels. Today, IoT and IoNT integrated systems coupled with Machine Learning (ML) and Artificial Intelligence (AI) promises to create high value and novel services that will transform the different domains already described.

It has been widely acknowledged that collaborative decision-making systems (DMS) have the potential to adapt services dynamically, though there are critical challenges that need to be addressed. Such challenges that hinder collaborative operations between devices, systems or services, include functional incompatibilities, limited built-in security, constraints in resources, trust, and transparency. For these reasons, most DMSs are centralized and depend on third-party services (e.g., Cloud services) where users have to submit and share their data with service providers and have no or limited access to and control of information. Consequently, this increases the risk of removing, tampering, and misusing the data. At the same time, centralized systems do not support end-to-end communications, although that is the key for automation of smart devices for operating timely services. On the other hand, highly dynamic data, and the static nature of most of the existing ML and AI algorithms, limit the capacity of processing data collectively, preventing the delivery of timely services. Therefore, these systems have critical risks such as data security threats, single point failure, and limitations in the scalability.

These challenges have, therefore, gained considerable attention on the practicality of centralized systems, and has motivated researchers towards restructuring them into fully decentralized systems enabling direct and secure inter-communication and inter-operation, minimizing the dependency on the third-party devices. Block Chain (BC) technology has become one of the promising solutions to build fully distributed systems, overcoming those challenges [2]. The BC technology has been widely used in the financial

- *D. Vimalajeewa and S. Balasubramaniam are with the Telecommunications Software and Systems Group, Waterford Institute of Technology, Waterford, Ireland, E-mail: dvimalajeewa@tssg.org and sasib@tssg.org.*
- *S. Thakur and J. Breslin are with the National University Galway, Ireland, E-mail : subhasis.thakur@insight-centre.org and john.breslin@nuigalway.ie.*
- *D. P. Berry is with the Teagasc, Animal & Grassland Research and Innovation Centre, Moorepark, Cork, Ireland, E-mail : Donagh.Berry@teagasc.ie.*
This work was supported by the Science Foundation Ireland (SFI) projects PrecisionDairy (ID: 13/1A/1977) and VistaMilk (ID: 16/RC/3835), and the Horizon 2020 GenTORE project.
E-mail: dvimalajeewa@tssg.org

sector, though its significance has not yet been fully realized in many other fields where numerous benefits may exist. Therefore, this gap is the key motivation for conducting this study, focusing on how the smart farming sector can be powered by BC technology to help achieve sustainable farming practices and effective food supply chain.

Smart farming (SF) is transforming farming practices by integrating modern ICT to achieve greater productivity through sustainable practices. IoT and IoNT devices deployed in SF can collectively monitor the farming process such as production quality and animal welfare. Collected data are then processed to derive insights which producers can use to make timely and informed decisions such as controlled delivery of farm inputs (e.g., quantity of feed, fertilizer) and early detection of disease. Since the collection of massive datasets from IoT and IoNT leads to exceeding resource capacities, limited scalability, and extended latency, cloud-based data processing methods were extended to Fog and then Edge computing. Such mechanisms facilitate data processing in close vicinity to the data sources and integrate derived insights for collaborative decision-making.

With the rapid expansion and advancement in SF, the main aim of SF is now, not only to intensify production but also manage an effective food supply chain. Cloud-, Fog-, and Edge-computing paradigms coupled with IoT and IoNT enable performing such operations effectively. However, several concerns have been raised with regards to these computing paradigms in that they may not be providing adequate support for addressing certain criteria effectively [3]. Most specifically, food quality, safety and trust in production process, along with transparency, auditability, and trustability of the practices have become an important factor with the involvement of various stakeholders along the food supply chain. For instance, the high demand for food has created greater market competition, and consequently has increased the availability of similar products for different brands. Therefore, consumers are now more conscious about production quality and brand trust for the products they purchase. Information for ensuring quality and safety standards is, however, often lacking. This is due to the data privacy and security issues and fraud behaviors, where producers are reluctant to share data such as the use of synthetic chemicals in the food production process for ensuring food safety and quality. Therefore, considerable attention has been expended on the urgency of restructuring the existing mechanisms that enable practices to be more transparent, auditable, secure, and traceable for the food supply chain with increasing contribution from as many as possible stakeholders such as producers, policymakers, and government bodies.

The contribution of this study is, therefore, to explore how IoNT, IoT, and BC can be integrated to develop a DMS based on an application that detects the level of chemical usage in farmlands. The use of different chemicals such as synthetic fertilizers (e.g., Nitrogen (N), Phosphorus (P), and Potassium (K)) and both herbicides and pesticides in SF is a common practice to maintain optimal soil quality, increase the yield and quality of crop and vegetables, control pest attacks and prevalence of diseases, and produce high-quality pasture. Today, a number of agricultural practices do not achieve these goals mainly due to the use of chemicals with insufficient monitoring and requirements. Consequently, these practices bring critical challenges such as farmlands becoming more vulnerable to depletion in soil fertility, reduced productivity, and potential residues. Therefore, having a mechanism to detect the soil fertility (i.e., soil available nutrient content) and then applying necessary fertilizers and other chemicals based on its need, is crucially important for maintaining optimal soil fertility for sustainable intensification of farm production. Therefore, the BC-based approach summarized below is the methodology we propose for identifying the chemical levels (or usage) in farmlands. It will not only be used for detecting chemical level but also used to make a platform for strengthening trust between producers and customers through chemical usage data that are available to customers as a color token. In addition, we introduce a token-based credit system to ensure the credibility and traceability of farms being compliant with chemical standards in their production processes.

a Explore the functionality and usability of affinity-based nano-sensors for sensing the availability of certain chemicals in soil.
b Incorporate data collected from IoNT sensors with IoT devices and then using the Bayesian probability updating approach for deriving insights regarding the chemical levels in the farm or crop.
c Insights from step [b] are combined with the BC technology to demonstrate the effectiveness of the BC-enabled approach for detecting the use of chemicals.
d Form a color token to represent chemical usage and securely share it through the BC network.

The remainder of the paper is organized as follows. Section 2 discusses related works and Section 3 explains the system model. In Section 4, the system model described in Section 3 is applied to simulated data to detect the level of chemicals. Section 5 discusses an approach for ensuring traceability and credibility of the proposed BC system, and Section 6 concludes the paper.

2 Related Work

This section first discusses IoNT and IoT networks, and then emphasizes the significance of incorporating the BC with these networks in various applications. Second, the BC mechanism and its various applications are explored. Finally, the significance of the present study is discussed with respect to the already existing solutions.

2.1 IoNT and IoT based networks

A nano-network is formed by interconnecting a large number of miniature-size IoNT devices. In such networks, data sensed by nano-sensors are aggregated at nano-routers to send to nano-micro interfaces which can consolidate such data and then communicate with both the nano- and macro-scale devices (IoT). Communications in such networks are mostly performed by molecular and electromagnetic communication methods which respectively communicates between IoNT and nano- and macro-scale devices [1]. Since these IoNT devices can, however, perform limited tasks due to limited resources, they are usually operated in integration with IoT-devices. Therefore, IoNT integrated with IoT

improve the potential of enhancing, and also scaling up, the services of existing systems such as controlled delivery of drugs.

Currently, such integrated networks based DMSs mostly operate under Cloud-computing paradigm [3]. Extended latency, network congestion, and safety of data are some of the critical problems that limit the use of this computing method in many applications. Alternatively, distributed computing approaches based DMSs like Fog- and Edge-enabled systems coupled with state-of-the-art ML techniques such as federated learning and deep learning have been proposed. Such systems can effectively overcome those issues while ensuring data privacy and security, thus enhancing the reliability and timeliness of outcomes [4].

However, trustability, integrity, and functional incompatibility are some of the most important factors which hamper the interoperability of such advanced computing systems, technologies, and devices. Consequently, those computing systems operate in isolation and are poorly scalable, so that their full potential, as well as the collected data, are significantly under-utilized. Therefore, the urgency of developing alternatives to handle these issues has gained considerable attention from the wider research community. A number of recent studies such as [5], [6], [7] have emphasized that BC and IoT are growing together and codependent, so that they have a greater potential of overcoming such issues.

2.2 Block Chain(BC)

The distributed ledger technology (DLT) is a fully decentralized peer-to-peer (P2P) method used for recording transactions (i.e., data) in an immutable ledger, with the mechanism for processing, validating, authorizing transactions. Block chain (BC) is an application of DLT for securely storing data and also known as *'internet of value'* [7]. BC generally consists of three components; blocks containing transaction data, a P2P network for direct communication, and a shared ledger for distributed data storage. Each block contains data with a hash value (a unique identification key) and a pointer to the hash of the previous block. A consensus algorithm is used for creating a new block and then appending it to the ledger. The Proof-of-Work (PoW) mechanism is the most commonly used consensus algorithm [2], [8]. In PoW, a hash key is allocated to a block through a mathematical puzzle, which is hard to solve, but easy to verify. Deriving a unique hash key is a computationally heavy task so that nodes which have sufficient resources are used for that and known as mining nodes.

Today, different versions as well as types of BCs are available for various applications. The study [8] stated two versions of BC as BC.01 and BC.02, which uses cryptocurrency and smart contracts (explain later), respectively. There are mainly three types of BC termed consortium-, private-, and public-BC [7]. While the consortium-BC is a group of BCs which has a set of members who can control the BC [9], a trusted central entity controls the private BC and termed permissioned BC [10]. The public-BC is called permissionless BC because there is nobody to control it and also anybody can join (or leave) the BC network at any time. Selection of the most suitable BC depends on the application requirements though the study [8] emphasized that there are unique properties such as irreversibility, traceability, anonymity, security and transparency that enable the use of BC in a vast range of applications. Some of their most prominent applications are intelligent management [11], smart transportation [12], agriculture [5], and healthcare [13].

2.3 Integration of IoT with BC

Many attempts have been made to incorporate BC technology with IoT-based systems. The intention of forming such integrated systems can be mainly seen under three categories: overcoming the issues in existing IoT-based platforms, optimizing the resources for efficient BC operations, and testing usability and improving the reliability of DMSs in different applications.

Considering the studies conducted to overcome the issues of the existing IoT-based platforms, the study [4] proposed a BC-enabled federated learning (*BlockFL*) method to overcome general issues in federated learning-based distributed ML systems such as single point failure. This study also investigated end-to-end learning competition aiming to find optimal block generation rate. Meanwhile, an analytical model proposed in [14] discussed the optimal deployment of full functional BC nodes for a BC-enabled wireless IoT system, minimizing data security issues. Finding sufficient resources for employing BC-powered IoT systems is a critical challenge that has been taken into a broader consideration. Data management and access control methods for BC-enable IoT systems given in [15] explained how time-series data could be stored at the edge of the IoT network for effective processing. The study [16], for instance, proposed an auction-based resource allocation method in connection with Cloud/Fog computing while a BC-based big data-sharing platform for resource-limited edges was developed in [17] by considering the challenges in deploying BC in edge devices. With regards to enhancing the performance of BC systems, BC-enabled edge computing approach was proposed in [10], aiming to ensure data privacy and energy security for power smart grid network. The work presented in [9] also explains how to manage power for plug-in electric vehicles in smart grids. Moreover, consortium BCs have been widely used for enhancing BC performance. The study [13] discussed a consortium BC-based mechanism for improving the accuracy and effectiveness of disease diagnosis in health-care.

However, some studies warned that these integrated systems could have unfavorable responses. The study [7], for instance, warned that this integration could also create unnecessary computational overhead and may not generate any tangible benefits. Therefore, [11] and [7] recommended to conduct an initial case study to make sure that integration with BC is necessary, proposing a checklist to conduct such a feasibility study.

The evidence alresdy provided, however, emphasizes the significance of BC-powered IoT systems, highlighting the performance of the existing IoT systems which can be empowered with improved scalability. On the other hand, it is already proven that IoNT can contribute to improving the performance of the IoT-based systems. However, there is no evidence that any attempt has been made so far to

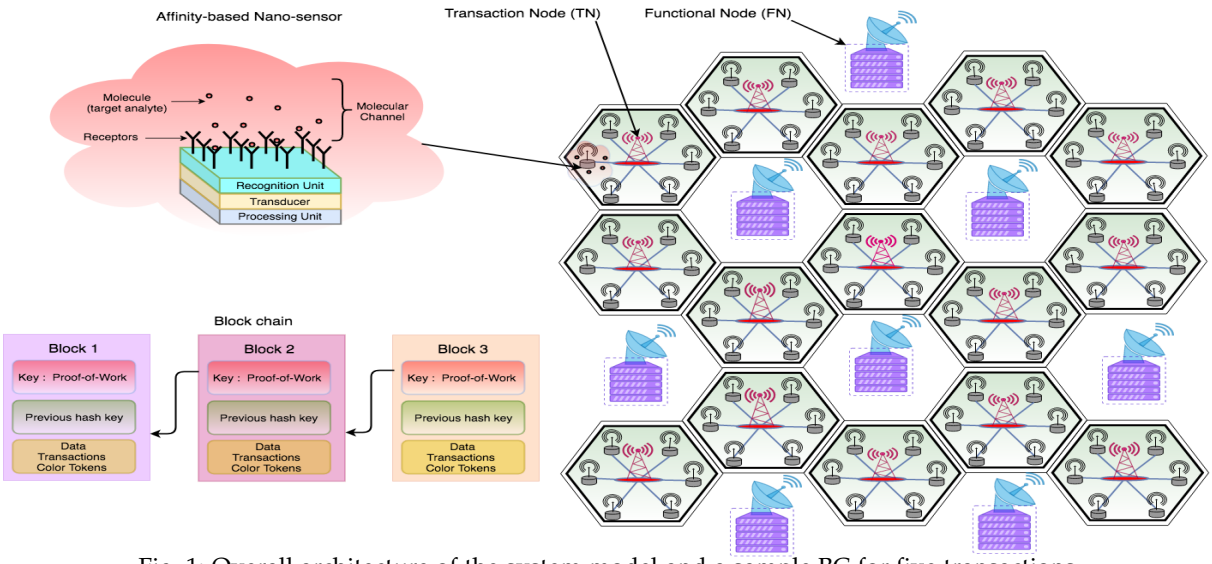


Fig. 1: Overall architecture of the system model and a sample BC for five transactions.

integrate IoNT into BC-powered IoT systems. Therefore, this study aims to fill that gap by exploring how IoNT and IoT network can be powered by BC mechanism. The next section presents a system model to explain how IoNT can be used in BC-powered IoT system.

3 System Model

The system model presented in this section is an integration of the functionalities of four components which includes: IoNT sensors, IoT devices (gateways-TN), miners (FN), and the BC network. Figure 1 illustrates the system model architecture, including IoNT sensor and an example BC network. The IoNT sensors are connected to each gateway, and it is assumed that IoT devices are connected with each other as well as with the FNs. The functionality of each component in the system is explained in the following subsections.

3.1 Affinity-based IoNT sensors)

Among the different types of IoNT sensors, affinity-based electrical bio-sensors are the most commonly used type for monitoring the presence of a specific chemical in a medium. The selective binding between molecules in a chemical sample and target analytes is the primary mechanism used for detecting the presence of that chemical [1]. The role of the affinity-based nano-sensors is, therefore, to facilitate binding target analytes with bio-molecules (known as receptors), which are functionalized on the nano-sensor surface. As a result, the sensor generates a signal reflecting the abundance of target analytes as a variation in the voltage. Based on the affinity sensor illustrated in Figure 1, when target analytes bind with the receptors, the following sequence of operation occurs:

1. The recognition unit selectively detects the target analytes.
2. The transducer converts the recognized events to processable signals in the form of electrical pulses.
3. The processing unit extracts the insights encoded in the signal.

The next section explains the procedure for deriving insights, which is the analyte concentration from the sensor signal.

3.1.1 Deriving analyte concentration through affinity-based nano-sensors signal

The Langmuir model [18], which explains a simple $1:1$ interaction between two molecules, is used to describe the functionality of the affinity sensor. The model assumes that all binding sites are equivalent and independent. When the binding interaction occurs between an analyte A (i.e., target molecule) and a receptor molecule B, it forms a chemical complex C with a k_a binding association rate and disassociating rate k_d. The chemical reaction between A and B is represented as $A + B \rightleftharpoons C$. Since the number of receptors ($[B]$) is fixed on the sensor surface, the concentration of A is proportional to $[C]$. This means that a change in the sensor signal (R) is proportional to the concentration of C and, therefore, the concentration of A. In the present study, R is computed in response units (RU), assuming $[B] = B_{max}$ is fixed [19]. The procedure for deriving the formula for extracting information about the analyte concentration (i.e. $[A]$) is explained below in four steps.

1. Considering the first order kinetics of the chemical reaction between A and B, the rate of change in $[C]$ and $[A]$ can be written as follows;

$$\frac{d[C]}{dt} = k_a[A][B] - k_d[C] \quad \text{association,} \quad (1)$$

$$\frac{d[A]}{dt} = -k_a[A] - k_d[C] \quad \text{disassociation,} \quad (2)$$

where k_a and k_d are the association and disassociation rates, respectively.
As $R \propto [C]$, the maximum R, $R_{max} \propto B_{max}$ at any time t, will result in free receptor concentration

$[B] = B_{max} - [C]$, i.e., $R \propto R_{max} - R_a$. Therefore, the equations can be re-written as:

$$\frac{dR_a}{dt} = k_a [A] (R_{max} - R_a) - k_d R_a, \quad (3)$$

$$\frac{dR_d}{dt} = [A] - k_d R_d. \quad (4)$$

2. The expressions for sensor response R in the association and disassociation phases (R_a and R_d) are derived by analytically solving the two differential equations 3 and 4 (note: $[A] = 0$ in the disassociation phase).

$$R_a = \frac{k_a [A] R_{max}}{(k_a [A] + k_d)} \left(1 - e^{-t(k_a [A] + k_d)}\right), \quad (5)$$

$$R_d = R_{d0} e^{-k_d t}, \quad (6)$$

where R_{d0} is the level of signal at the end of the association.

3. When $A + B \rightleftharpoons C$ reaches its equilibrium state, R_a is at its maximum (say R_{eq}). Therefore, R_{eq} is derived from 5 as:

$$R_{eq} = R_{max} \frac{[A]}{[A] + k_D}, \quad (7)$$

where $k_D = \frac{k_d}{k_a}$ is the affinity constant.

4. The response factor (RF), which represents R_{eq} relative to R_{max}, is computed as:

$$RF = \frac{R_{eq}}{R_{max}} = \frac{[A]}{[A] + k_D}, \quad (8)$$

where $k_D = [A]$ and $RF = 0.5$. This means 50% of receptors (i.e.,B molecules) are occupied by A molecules at the equilibrium stage. This reaches 100% when the sensor becomes saturated.

To fit a response curve (i.e., RF model), the RF formula (8) is used for computing RF values for a range of $[A]$ values. The RF model represents the relationship between RF and $[A]$ and enables identification of the sensor response region where a significant change in RF can be obtained in response to the change in $[A]$. Also, this model can be used to derive the $[A]$ corresponding to any RF value. Finally, this RF value is sent to a gateway node to which the nano-sensor is connected to. Algorithm 1 presents the process of computing RF for a given $[A]$. A detailed discussion about deriving the RF model can be found in [18] and [20].

Algorithm 1: Nano-sensor Response Factor (RF)

Input : k_a, k_d, R_{max}, & $[A]$
Output: RF

Initialization for the association phase ;
$R_{(a,t=0)} = \epsilon, t = 0,$ and $\varepsilon_R = 1 \times 10^{-5}$
while $\epsilon \geq \varepsilon_R$ **do**
  $R_{a,t} = R_a(t, k_a, k_d, [A])$
  $\epsilon = |R_{a,t} - R_{a,t-1}|$
  $t = t + 1$
end
Initialization for the disassociation phase ;
$R_{d,t} = R_{a,t}, t = t,$
while $\epsilon \geq \varepsilon_R$ **do**
  $R_{d,t} = R_d(t, k_d)$
  $\epsilon = |R_{d,t} - R_{d,t-1}|$
  $t = t + 1$
end
$RF = \text{Max}\{R_{a,t}, R_{d,t}\}/R_{max}$

TABLE 1: Response Classes (RC)

RF (%)	<20	20-40	40-60	60-80	>80
RC	A	B	C	D	E

TABLE 2: One-way response class (RC) frequency ($f_A = \sum_{i=1}^{N} f_{A,i}$)

RC \ Node	1	2	⋯	N	Total
A	$f_{A,1}$	$f_{A,2}$	⋯	$f_{A,N}$	f_A
⋮	⋮	⋮	⋱	⋮	⋮
E	$f_{E,1}$	$f_{E,2}$	⋯	$f_{E,N}$	f_E

3.2 IoT Sensor Node

In the present study, it is assumed that there is one gateway per farm as illustrated in Figure 1. The gateway processes data transmitted from nano-sensors within its range which it sends to the mining nodes, while also storing information from other mining nodes on the level of chemicals within the region.

3.2.1 Data Processing Operation

The chemical-detecting indexes such as N, K, and P have been defined as classes that consists of ranges. For example, the recommendation for P and K indexes are between 3 and 4 in order to maintain optimum soil fertility level [1]. Our study categorizes RF data collected by each gateway node from its nano-sensors into five classes termed Response Classes (RCs). RCs are defined by dividing the sensor's RF range (i.e., 0-100%) into five non-overlapping regions, as presented in Table 1. Therefore, they correspond to the fixed ranges of $[A]$ in the sensor active region. This is followed by computing the frequency of the five RCs in each gateway using the RF values gathered over a period of time, resulting in Table 2. This table is referred to as a *one-way frequency table* as it enables the derivation of conditional one-way relative frequency (i.e., marginal probability) by dividing each row by its sum of row frequency values. The relative frequency values are called conditional probabilities. That is, for instance, $P(1|A) = f_{A1}/f_A$ is the probability of the gateway node-1 being in the response class A. Algorithm 2 summarizes the process of computing the relative frequencies for period t. We are, however, interested in the probability of a selected gateway being in a particular RC (i.e., $P(A|1)$- inverted value of $P(1|A)$). Therefore, these conditional probabilities are fed into the Bayesian updating method explained in the next section to compute those probabilities.

3.2.2 Sequential Bayesian Updating

In this section, Bayesian theory is briefly introduced and then the Sequential Bayesian Updating (SBU) method is

[1]https://www.teagasc.ie/crops/soil--soil-fertility/soil-analysis/

Algorithm 2: Computing $P(TN_{ID}|\Theta)$

Input : TN_{ID}s and # of nano-sensors (K)
Output: $P(TN_{ID}|\Theta)$

foreach $i \in \{N\}$ **do**
  foreach $j \leftarrow 1 \quad to \quad K$ **do**
    $\{RF_j\}_i =$ **Algorithm 1**
  end
  $[f_i]_{5\times 1} \leftarrow$ assign RC labels to $\{R_f\}_i$ and calculate their frequency
  collect f_is into $F_{5\times N}$
end
$P(TN_{ID}|\Theta) = F_{i\times N}/\sum_{j=1}^{N} F_{i,j} \quad$ for $\quad i = 1, \cdots, 5$

discussed for computing the probability distribution of any selected gateway node (i.e., farm) being in the five RCs.

Bayesian theory is commonly used in statistical inferences as it allows updating the inverted conditional probability based on the latest collected data/evidence. We assume there are N gateway nodes and two random variables, which are the selection of a gateway node ($\mathbb{G}$) and a RC (Θ). For each of these random variables, their sample spaces are then $\mathbb{G} = \{1, 2, \cdots, N\}$ and $\Theta = \{RCs\}$. If a node $i \in \mathbb{G}$ is selected, the probability of the selected node being in the $j^{th} \in \Theta$ RC is computed using the Bayes theory as represented as follows:

$$P(j \in \Theta | i \in \mathbb{G}) = \frac{P(i \in \mathbb{G} | j \in \Theta) P(\Theta)}{P(\mathbb{G})}, \tag{9}$$

where $P(\Theta)$ is known as the prior probability distribution function (PDF) and represents the strength of the belief of a node being in the five RCs. The likelihood of the result given the prior distribution is represented by $P(i \in \mathbb{G} | j \in \Theta)$, where $P(\mathbb{G})$ is known as the evidence (or data) and computed as $\sum_{\forall j \in \Theta} P(i \in \mathbb{G} | j \in \Theta)$. Applying this formula for all gateway nodes, the probability of each node being in the five RCs $[P(j|i)_{i=1}^{N}]_{j=1}^{5}$ can be computed to produce the matrix $P(\Theta|\mathbb{G})_{5\times N}$. This matrix is known as the posterior PDF computed based on the set of RC frequency data samples over a time period T.

In the SBU, when a new RC frequency data sample is collected over the time period $(T+1)$, the new posterior PDF is computed by using $P(\Theta|\mathbb{G})_{5\times N}$ as the prior probability matrix, as is represented as

$$P(\Theta|\mathbb{G})_{(T+1)} = \frac{P(\mathbb{G}|\Theta)_{(T+1)} P(\Theta)_{(T)}}{P(\mathbb{G})_{(T+1)}}, \tag{10}$$

where $P(\Theta)_{(T)} = \left[\Pi_{k=0}^{(T)} P(\mathbb{G}|\Theta)_k\right] P(\Theta)_0$. Based on this, the probability of the i^{th} node being in the five RCs over the time period $(T+1)$ can be computed as

$$P(\Theta|i)_{(T+1)} = \frac{P(i|\Theta)_{(T+1)} P(\Theta)_{(T)}}{P(i)_{(T+1)}},$$

where $P(\Theta)_{(T)} = \left[\Pi_{k=1}^{(T)} P(i|\Theta)_k\right] P(\Theta)_0$.
The $P(\Theta|\mathbb{G})$ updating process will continue until there are changes in the posterior PDF that reaches a certain threshold. Algorithm 3 summarizes the SBU steps. The optimal number of updating steps required for detecting the level of chemicals in a farm will be discussed in detail in the next section.

Algorithm 3: Computing the posterior probabilities using SBU

Input : **Algorithm 2** & $P(\Theta)_{T-1}$
Output: $P(\Theta|TN_{ID})_T$ and $P(\Theta)_T$

Initialization: $\epsilon = 1 \times 10^{-5}$
while $d \leq \epsilon$ **do**
  $P(TN_{ID}|\Theta)_T \leftarrow$ execute Algorithm 2
  `/* relative frequency */`
  foreach $i \in \{TN_{ID}\}$ **do**
    `/* Posterior Probability, PoS_{5×N} */`
    $P_i = f_i * p_i$, where $f_i \in [P(TN_{ID}|\Theta)_T]_{(5\times i)}$ and $p_i \in [P(\Theta)_T]_{5\times i}$
    if $\sum P_i == 0$ **then**
      $PoS_i = 0$
    else
      $PoS_i = P_i / \sum P_i$
    end
    collect P_is into a matrix $PoS_{5\times N}$
    `/* Update Prior Probability, PrI_{5×N} */`
    Indexes k of f_i where $f_i == 0$
    if *number of* $k > 0$ **then**
      Replace entries of P_i at k indexes by the values of p_i such that $P_i[k] = p_i[k]$
    end
    collect P_is into a matrix $PrI_{5\times N}$
  end
  $P(\Theta|TN_{ID})_T = Pos$ and $P(\Theta)_T = PrI$
  $d = \sum |P(\Theta|TN_{ID})_T - P(\Theta|TN_{ID})_{T-1}|$
  $T = T + 1$
end

3.3 Mining/Functional Nodes

It is assumed that mining nodes are trustable entities and have the authority to control the BC. The mining nodes also use the Joint Cloud service for performing block mining [21]. In the Joint Cloud service, the Cloud service providers may consist of government bodies such as an Agricultural Department Agency or authorized pharmaceutical companies.

3.3.1 Joint Cloud and incentive for mining

The use of IoT sensors in SF is a business model for sensor Cloud. In this model, the Cloud provides IoT as a service by collecting and aggregating IoT-sensor data. The Cloud provider can place the sensors in the farms and collect the sensor data for the regulators (miners in the present study) for verification and traceability. In this business model, the farms will outsource the IoT data collection and report the job to the IoT Cloud service. This IoT sensor as a service model can be implemented using BCs following the two steps given below.

1. BC mining can be used as an incentive for proving the IoT sensor service. For example, in a PoW-based BC system, the IoT sensor service provider (i.e., miner) can gain benefits by collecting mint tokens as it produces new blocks.
2. The IoT service is a geolocated service, where the quality of service depends on the location of the sensors and the cloud to reduce latency. The geolocated property supports joint cloud system. In this system, there will be multiple IoT cloud providers and they collaborate to provide a chemical traceability service for the regulators. In this interaction model between the service providers, BCs are very useful as IoT cloud providers that do not have trust between each other.

The Clouds process the data from each farm (i.e., gateway) and offers certification through a colored token, which represents the level of chemical used on the farm. They also perform credit and tokens transactions between the farms and the regulators to offer (or charge) a certain amount of credits based on the quality of products and tracking the traceability, respectively. We will first describe the process of credit exchange and this will be followed by a description on the full functionality of the mining process of each nodes.

3.3.2 Farm credits

Let us assume that Cr_T amount of credits is assigned by a governmental agency or miners to each farm when it joins the BC network at $T = 0$. For each mining step, the credibility is computed following the three steps given below.

1. Farmers are rewarded or penalized a certain amount of credits for either complying or not complying with the chemical standards.
2. A farm is compliant with the chemical standards if it has a higher probability (i.e., at least of not being in the class E (i.e., $P(\sim E) \geq .8$),
3. The rate of change in the amount of remaining credits after several mining steps represents the credibility.

For any farm, the amount of credits at the T^{th} mining step is computed as:

$$Cr_T = \begin{cases} Cr_{T-1} - P_{(T,E)}L + P_{(T,\sim E)}M & f_{(T,E)} \neq f_{(T-1,E)}, \\ Cr_{T-1} & \text{otherwise}, \end{cases} \tag{11}$$

where, $L = e^{\alpha f_{(T,E)}}, M = e^{\alpha f_{(T,\sim E)}}$ and $f_{(T,E)}$ is the cumulative frequency of a farm being in the class E up to the T^{th} time step, $P_{(T,E)}$ is the posterior PDF of that farm being in the class E at the T^{th} mining step, and $\alpha \in \mathbb{R}$ is a constant and termed as the credibility tuning parameter.

The amount of credits reduced or reward will exponentially increase if there is a change in the frequency of the response class E. That is, if $f_{(T,E)} \neq f_{(T-1,E)}$), then $P_{(T,E)}e^{\alpha f_{(T,E)}}$ amount of credits is reduced from the available credits Cr_{T-1} of farms which belong to the class E, while the amount of credits is added to the farm for not being in the class E is $P_{(T,\sim E)}e^{\alpha f_{(T,\sim E)}}$. The amount of credits remains the same (i.e., $Cr_T = Cr_{T-1}$) if there is no change in frequency of being in the class E.

3.3.3 Block Chain Network

This section presents how the BC is incorporated with the SBU, joint Cloud system, and the credit computing approach for detecting the level of chemicals used on the farms. The BC network employed here is a private BC. The gateway nodes and the mining nodes in the BC network are termed as the transaction nodes (TN) and functional nodes (FN), respectively. The TNs collect data and send them to a selected FN which aggregates the TNs' data and mines two blocks for the TN and FN networks (more details are given below). Furthermore, data communicated between the TN and FN networks are encrypted in order to protect data privacy, security, and integrity.

The functionality of the BC network uses three steps; data sharing, data processing and block mining, and BC updating, and each of these steps are described as follows (Algorithm 4 summarizes these steps).

1. **Data sharing:** Algorithm 2 is executed at each TN for collecting RC frequency data samples for a period of time T (this is termed as data stream). Each data stream is then encrypted and access permissions are granted by using the authenticated symmetric encryption under the *Advanced Encryption Standard Galois Mode* (AES-GCM)[2] and using the Shared Secret Key (SSk). The encrypted data streams are then submitted to a selected FN, which is the miner.
   The AES-GCM enables decryption of data by using the same SSk key that is used for encrypting the data. This encryption generates compressed data in plain-text which contains a key value used for integrity protection and authentication at the decryption. Hence, any FN that contains the encrypted key can verify the integrity of the encrypted data and perform an authenticated decryption. Each TN will share the encrypted data and the its Public Key (PK) with a selected FN.
   The PK and secret key (SK) for each TN is generated using the *Diffie-Hellman* (D-H) key exchange service, which facilitates the sharing of a common secret key between two or more parties. The SSk at a TN is generated by using its SK and PK of the FN through the D-H service. This is one of the reliable techniques for sharing data between unknown parties as the D-H service avoids sending the SK away from its owner while also allowing the owner to revoke sharing data at any time [3].
2. **Data processing and block mining:** Prior to creating a block at the FN, the data are processed as follows:
   a. Data coming from a TN is decrypted through the SSk generated using the FN's SK and the PK of the TN included in the data.
   b. A ML task is executed as a smart contract to derive insights about the level of a chemical (this ML task if further described below).
   c. By using the insights derived in the previous step, a color token is generated to represent

[2]https://cryptography.io/en/latest/hazmat/primitives/aead/
[3]https://cryptography.io/en/latest/hazmat/primitives/asymmetric/dh/

Algorithm 4: Block mining

Input : parameters required for **Algorithm 1,2**, and **3**, and Tw, α
Output: block

Initialization: Genesis block containing $P(\Theta)_0$ and Cr_0
foreach $T \in$ # *of block mining steps* **do**
/* select a FN and generate its SK and PK */
$FN_k \leftarrow$ select any id from $\{FN_{id}\}$
$SK_{(FN,k)}, PK_{(FN,k)} \leftarrow$ D-H key exchange service
/* Extract the prior probability from the latest block */
if *T = 0* **then**
$P(\Theta)_{T-1} \leftarrow$ from the genesis block
else
$P(\Theta)_{T-1} \leftarrow$ from the last block of the BC of the FN_k's ledger
end
/* Collecting and sharing data at TNs */
$\forall i \in \{TN_{id}\}_{i=1}^{N}$
$SK_{(TN,i)}, PK_{(TN,i)} \leftarrow$ D-H key exchange service
$SSk_{(TN,i)} \leftarrow$ D-H key service$(SK_{(i)}, PK_{(FN,k)})$,
collect RC frequency $\{F_i\}_{t=1}^{T} w \leftarrow$ execute **Algorithm 2**,
$a_{i,T} \leftarrow$ AES-GCM encryption$(\{F_i\}_{t=1}^{T} w, SSk_i)$,
FN_k collects $Data_{(i,T)} = [a_{(i,T)}, PK_{(TN,i)}]$.
/* Processing data and block mining at FN */
$\forall i \in \{TN_{id}\}_{i=1}^{N}$,
$PK_{(TN,i)} \leftarrow$ get from $data(i, T)$,
$SSk_{(FN,k_i)} \leftarrow$ D-H key service$(SK_{(FN,k)}, PK_{(TN,i)})$,
$b_{(i,T)} \leftarrow$ AES-GCM decryption$(a_{(i,T)}, SSk_{(FN,k_i)})$,
collect$b_{(i,T)}$toF_t.
$P(\Theta|TN_{ID})_T \leftarrow$ execute **Algorithm 3**
foreach $i \in \{TN_{id}\}_{i=1}^{N}$ **do**
Generate color token $(CT_{(i,T)}) \in \leftarrow [P(\Theta|TN_{ID})_T]_{5\times i}$
Update $[f_T]_{(T,E)}$
Credits $(Cr_{(i,T)}) \leftarrow Cr_{(i,T-1)} + \left[P(\Theta|i)_{(T,E)} e^{\alpha[f_i]_{(T,E)}}\right] + \left[(1 - P(\Theta|i)_{(i,\sim E)}) e^{\alpha[f_i]_{(T,\sim E)}}\right]$
collect $\left[CT_{(i,T)}, Cr_{(i,T)}\right]$ into $Data_{(T)}$
/* mine TN block */
$Data_{(i,T)} \leftarrow$ AES-GCM encryption($\left[CT_{(i,T)}, Cr_{(i,T)}\right], SSk_{(FN,k_i)}$)
mine block for TN network adding $[D_{(i,T)}, PK_{(FN,k)}]$
end
/* mine FN block */
mine block for FN network adding $[D_{(T)}]$
/* Updating ledgers */
$\forall i \in \{TN_{ID}\}$ and $\forall k \in \{FN_{ID}\}$, update ledgers
$tn \in TN_{ID}, SSk_{tn} \leftarrow key_{tn}(PK_{FN})$
end

the level of a chemical used. The color token represents the variability in chemical levels over a period of time T.

d. Based on the color token generated for each farm, the amount of credits held by each farm is updated.

Smart contract/ML task: The SBU updating process presented in Algorithm 3 is executed to derive the updated posterior PDF. Then for each TN, a color token is derived from its posterior PDF. A color token consists of five unique colors that correspond to the five RCs. The region occupied by each color in the color token is proportional to the probability of the level of a chemical being in the five RCs (explained in detail in the next section).

Since the TNs and FNs play two different roles in this system, the information required to be stored in the TN and FN networks is different. Therefore, two

blocks are created for the FN and TN BC networks. The block created for the FN network includes color tokens and credit transactions of all TNs because they are required to perform future block mining. When creating a block for the TN network, the color token and credit value of each TN are encrypted together as they are sent away from the FN network. The AES-GCM and H-D methods explained above are used for the encryption. The new block is then created, including the encrypted data and FN's PK. Finally, the FN block is added to the FN ledger while the TN block is sent to at least one TN to add it to the TN network. The PoW technique [2] is used for validation of the new blocks and then adding blocks into the BC ledgers.

3. **Updating BC:** After the TN's and FN's blocks are added to their BCs, all FNs and TNs update their ledgers accordingly. Any TN can derive the SSk by using its SK and FN's PK that is contained in the BC, which is also used for decrypting the data to have up-to-date information about the current level of chemicals used on the farm. Similarly, any FN can be a future miner as all necessary information required to perform a new mining task is contained in their BC. At the moment, the miner is selected randomly.

3.4 Performance metrics

Two performance metrics, *Mean Squared Error* (MSE) and *Accuracy* (AC), are used for assessing the performance of the BC-IoNT system. While MSE is used as the performance measure to decide how accurately the proposed BC system can detect the level of chemicals, the AC is the percentage of the number of TNs which the RC has identified correctly. These two metrics are computed as follows:

$$MSE = \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{N},$$

$$AC = \frac{\sum_{i=1}^{N} f_i}{N} \times 100, \quad f_i = 1 \quad \text{iff} \quad |y_i - \hat{y}_i| = 0,$$

where y and $\hat{y}$ stand for the actual and the predicted RC of which TN belongs to, respectively, and N is the number of TNs (i.e., farms).

4 Results

This section will evaluate the use of the proposed BC system for detecting the levels of chemicals in farmlands. First, experimental setups used for simulations are briefly explained. Next, the process of computing the RF values from the nano-sensor signals is discussed. The variability in the probability of each farm categorized in the five RCs is explained as the third step, and this is followed by discussion on selecting the optimal parameters for generating color tokens effectively. Finally, the color token is used to represent the levels of chemicals.

4.1 Experimental setups

It is assumed that the area covered by each gateway (TN) is the same and corresponds to a farm. However, in reality, a farm could contain a collection of such devices. The experimental procedure is as follows:

1. We assume that a chemical A is applied over the set of farms at different concentrations and this range is presented as $\mathcal{U}(0, 50)$. The reason for selecting this range is based on the k_a and k_d values and explained in detail in section 4.2. The distribution of $[A]$ on each farm was also varied by using the Gaussian distribution in response to different field conditions.
2. For a given period (T), the TNs collect a set of RC frequency data samples and then pass this to the FN, where a smart contract is executed to detect the levels of chemicals on the farms (i.e., $[A]$). The optimal number of RC frequency data samples required for deriving the level of chemicals in each farm with $p \leq 0.001$ accuracy is discussed later.
3. The last step is to perform credit transactions for each farm where two blocks are created.

These three steps are repeated for each application of chemicals. All studies were based on simulated data generated by using the parameters presented in Table 3. In the simulation, these parameters are used, unless mentioned otherwise based on the evaluation.

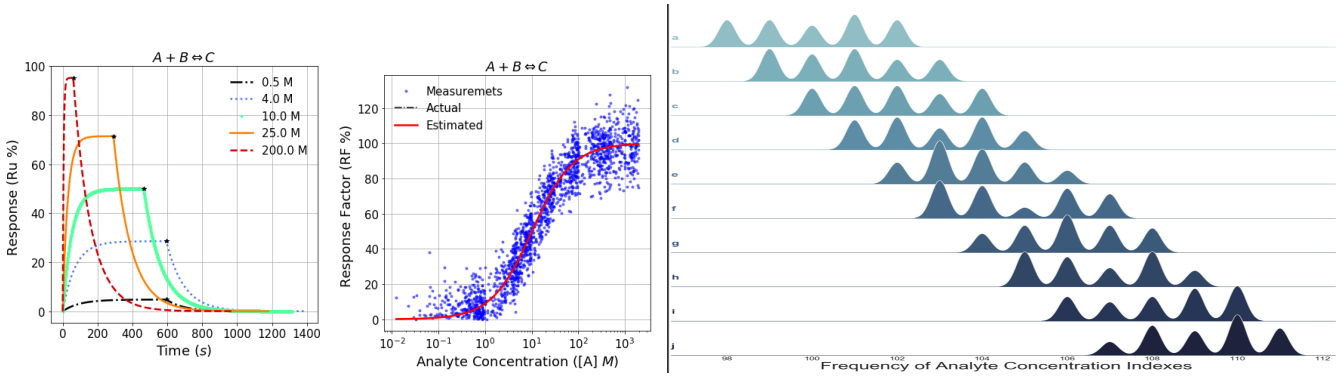
TABLE 3: Simulation Parameters

Parameter	Value
Association rate (k_a)	$10^{-2}(MS^{-1})$
Disassociation rate (k_d)	$10^{-3}(S^{-1})$
Receptor concentration (R_{max})	100 (Ru)
Number of Farms (N)	40
gateways (node) per field	1
Nano-sensors per node (K)	100
Analyte concentration ($[A]$)	$[\mathcal{U}(0, 50)]_{1\times N}$
Prior probability ($[P(\Theta)_0]_{(5\times 1)}$)	$[0.2]_{(1\times 5)}$
IoNT signal threshold (ε)	10^{-5}
Initial credits (Cr_0)	100

4.2 IoNT sensor response and response factor

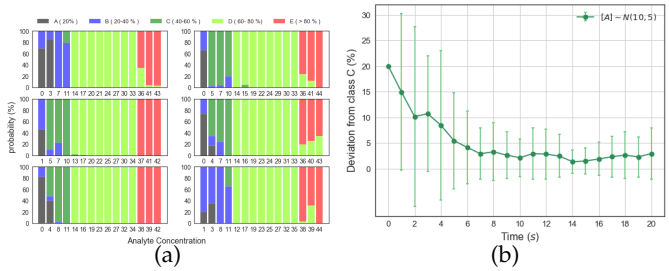
Algorithm 1 was executed for different $[A]$ values to study the variability in the sensor response (R) over time. Figure 2a presents the variability in the R for different $[A]$ values. With increasing $[A]$, the maximum R increases, while the time taken to achieve the maximum R reduces. In particular, when $[A] = 10M$ (i.e., $k_D = [A]$), the maximum sensor response is 50% and thus, it proves the theoretical fact mentioned in 8, which states that $k_D = [A]$, the $R_{max} = 50\%$ (i.e., half of the receptors are occupied by the molecules of A).

Obtaining information on the sensor active regions is important as it provides prior insights about the sensor capabilities. Hence, the sensor active regions was derived using two steps; (1) RF values were derived for a set of $[A]$ values in the range $[10^{-2}, 10^3]$ and then randomized by adding $\mathcal{N}(0, 1)$ error terms, and (2) a non-linear regression model was fitted to the RF values as the theoretical RF model in (8) is a non-linear function. The fitted RF model in Figure 2b indicates that the RF indicates a significant change in response to the change in $[A]$ within the range $10^{-1} \leq [A] \leq 10^2$. Therefore, this range was selected as the sensor active region and used to set up experiments for



(a) (b) (c)

Fig. 2: Nano-sensor response, (a) for different $[A]$ values, (b) for a range of $[A]$ values (i.e., response Factor model), and (c) frequency of RCs over a period of time.



(a) (b)

Fig. 3: Sequential Bayesian updating outcomes (i.e., probability) for detecting level of A, (a) when $[A]$ varied over the range $[0, 50]$, where $[A] = 0$ means $0 < [A] < 1$ and (b) deviation from the class C.

executing the proposed BC system. As shown in Figure 2b, the concentration ranges covered by the RCs from A to E are increasing. Hence, the average maximum level of $[A]$ was set as $50M$, aiming to obtain a fair distribution of RCs over the farms. Otherwise, the distribution of the RC could be concentrated towards the RCs corresponding to higher RCs (e. g., D and E) for the wider range of $[A]$.

4.2.1 *Response Class Frequency*

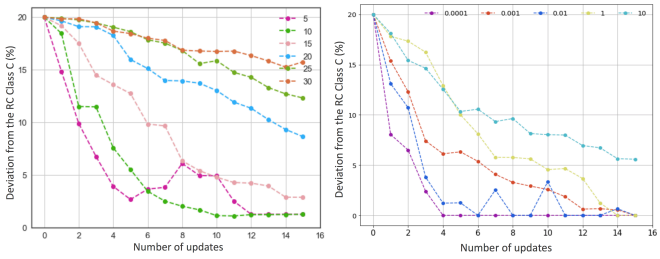
To illustrate the variability in the RC frequency data collected in Table 2, Algorithm 2 was executed at TNs for a period of time. Figure 2c shows the RC frequency distributions derived by using the kernel density estimation technique for ten TNs. The five peaks from left to right in each graph corresponds to the RCs from A to E. Based on the height of the peaks, the RC of each farm can be determined. However, the RC frequency distribution is a static measure and the level of chemicals could variy dynamically due to the influence of several time-variant factors such as land usage, weather, and prior chemical usage. Therefore, using only the RC frequency distribution is not sufficient enough to describe the availability of a certain chemical that has been used. Here, the next section discusses how the current and prior level of a chemical can be incorporated for dynamically updating the current levels of a chemical by using the SBU approach integrated with BC technology.

4.3 Probability of deviation from the optimal response class

This section first discusses the variability in the probability of the chemical levels on the farms and its position in the five RCs. This is achieved by randomly varying $[A]$ over the range $[0, 50]$ and then the probability of deviation from the optimal RC. The optimal RC was selected as the class C. In reality, this could be any of the RCs based on the application requirements.

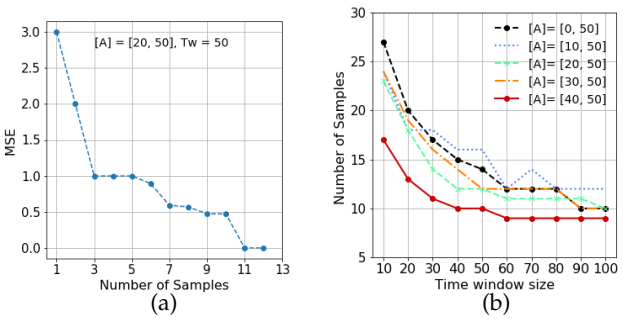
To compute the posterior PDF, the SBU method in Algorithm 3 was executed for several iterations, assuming that initially the level of a chemical in every farm has an equal chance of being in any of the five RCs (i.e., $P(\Theta) = [0.2]_{1\times5}$). Figure 3a, for instance, illustrates the variability in posterior PDF of $[A]$ in six farms that are within the five RCs with increasing $[A]$ for 15 probability updating steps. The probability of $[A]$ in farms being in the D and E RCs is greater for larger values of $[A]$ (at least ≥ 15), but when $[A]$ is below or around the equilibrium concentration (i.e., $[A] = 10M$), the farms belong to the A, B, and C RCs. Similarly, the same procedure was repeated for a number of iterations by taking into account the variability in $[A]$ of forty farms and computing the probability of deviation from the class C as the probability of not being in D in each updating step. Figure 3b shows that the deviation in probability reduces with increasing updating steps. Thus, by performing several probability updating steps, it can easily be recognized to which RC a selected farms chemical level is converging, thereby identifying the farms which are not compliant with the optimal chemical standards.

However, this convergence rate varies depending on various factors. Two of the most important factors are variability in $[A]$ within as well as among other farms and the affinity constant k_D. To illustrate their impact, the variability in the convergence rate of posterior PDF was explored for different variability levels in $[A]$ as well as a set of k_D values. The outcomes for this analysis are presented in Figure 4. The convergence rate was faster for smaller variability levels in $[A]$ within farms as well as smaller k_D values (i.e., larger association rates). Therefore, these outcomes unveil the criticality of deciding the optimal number of probability updating steps required for precisely deriving the RC of each farm.



(a) (b)

Fig. 4: The convergence of probability of deviation from the class C for different, (a) variability in levels of $[A]$ and (b) k_D values.



(a) (b)

Fig. 5: Selecting optimal number of samples required for detecting the level of $[A]$, (a) based on the variability in MSE and (b) with the increasing Tw and range of inter-farm $[A]$.

4.4 Optimal probability updating steps (sample frequency)

This section discusses selection of the optimal number of probability updating steps. That is, the number of samples (say S_{opt}) that each TN is required to send to FN based on the variability in $[A]$ within, as well as among other farms, and this includes the optimal sample frequency (i. e., optimal value T value termed as the time-window size (Tw)).

According to Figure 2a, the time taken to reach the maximum response varies with the variability in $[A]$. Waiting until all nano-sensor nodes reach their maximum response to collect sensor responses could create extended latency in detecting the level of $[A]$. This issue can be mitigated by collecting sensor responses within a fixed period of time (i.e., T_w). However, too small a value of T_w increases the S_{opt}, while larger T_w values extends the time for collecting RC frequency samples. Therefore, the derivation of optimal values for Tw and S_{opt} was conducted in two steps as follows:

1. Assuming that the variability in $[A]$ within farms is fixed (say $\sigma = 1$), the inter-farm variability in $[A]$ is fixed in step [a] and varied in step [b], respectively.
   a. When Tw is fixed to 50, to illustrate how precisely the system can detect the level of $[A]$, the behavior of MSE in detecting the RC of a set of farms was explored while varying inter-farm $[A]$ over the range $[20, 50]$. Figure 5a illustrates that the MSE reduces with the increasing number of samples. Under these settings, at least 12 samples are required ($S_{opt} = 12$) to detect the RC with $p \leq 0.001$ accuracy.
   b. Figure 5b exhibits the S_{opt} required for detecting the RC with increasing Tw size and decreasing inter-farm quantity of $[A]$. It can be seen that S_{opt} is decreasing while increasing Tw and decreasing inter-farm variability range of $[A]$. Therefore, this outcome confirms that when the Tw is large enough and variability in the inter-farm quantity of $[A]$ is less, the level of $[A]$ in a farm can be decided effectively with a fewer samples (or updating steps). This means that the convergence is faster.
2. When $[A]$ is varied within as well as among other farms, Figure 6 exhibits the behavior of the MSE and S_{opt} with intra-farm variability in $[A]$ (denoted as $[A]$ variance) and Tw size. Although the MSE becomes smaller with decreasing variance in $[A]$ regardless of the Tw size (Figure 6a), the corresponding S_{opt} is higher for smaller Tws (Figure 6b). Furthermore, Figure 6c depicts that the optimal Tw size and S_{opt} become higher with larger inter-farm as well as on a specific farm variance in $[A]$. This means that the time taken to detect $[A]$-level with $p \leq 0.001$ accuracy is longer.

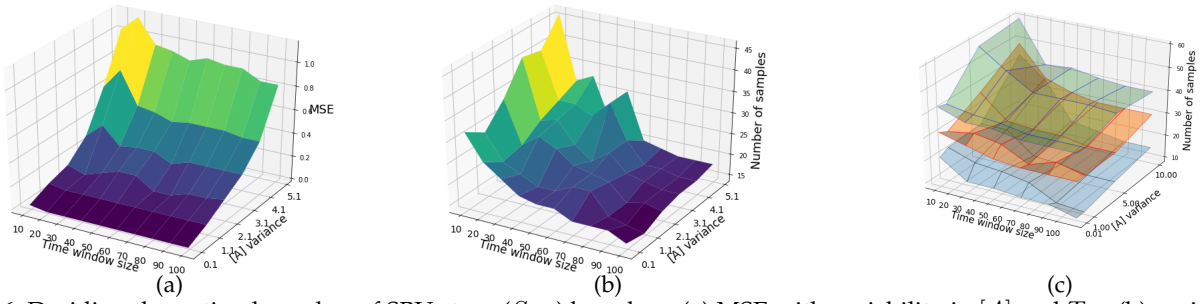
Therefore, Figure 5 and 6 reveal that if the system takes longer period at a slow convergence rate for detecting the levels of chemicals, it is an early indication that the variability in the level of that chemical is higher within and/or among farms which are connected through the BC system.

4.5 Color Tokens and BC System Performance

We assume in this analysis a chemical A was applied several times at a value of ten over forty farms. The average $[A]$ applied over the farms varied randomly within the range $[0, 50]$. Since the average $[A]$ over each farm could vary due to the variability in field conditions such as land usage, it was assumed that the amount of this variability randomly changes over the range $[0, 5]$ (i.e., variability in $[A]$ over a farm is the average $[A] \pm p$, where $p \pm \mathcal{U}(0, 5)$). Values for Tw and S_{opt} were selected as 50 and 15, respectively, because under similar experimental settings, the previous section showed that with these values, the level of $[A]$ can be detected effectively with $p \leq 0.001$ accuracy.

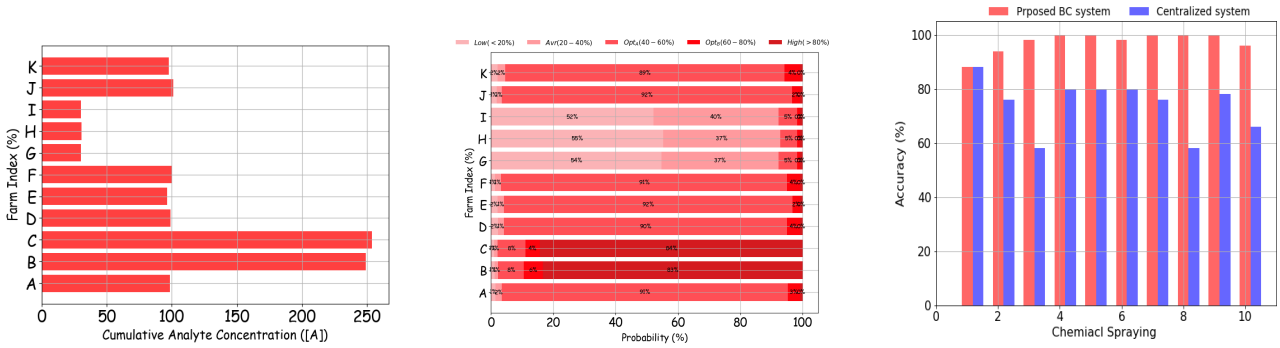
At each application of A, Algorithm 4 was executed to derive the probability of $[A]$ in each farm belonging in the five RCs. The probability values were then converted into a color token. For instance, Figure 7a and 7b illustrate, respectively, the total amount of chemicals used and the corresponding color token created for ten selected farms. The color token corresponding to the farms B and C, which have used a higher level of chemicals, indicates greater probability ($> 80\%$) of being in the RCs D and E. On the other hand, farms which have used least, for instance, G, H, and I, are mostly limited to the RCs A and B. Therefore, the color token is a good indicator to represent the levels of the chemicals in the farm.

In addition, the accuracy of detecting the RCs of all farms was also computed by using the proposed BC-IoNT



(a) (b) (c)

Fig. 6: Deciding the optimal number of SBU steps (S_{opt}) based on, (a) MSE with variability in $[A]$ and Tw, (b) variability in $[A]$ and Tw, and (c) MSE with variability in $[A]$; top - $[0, 50]$, middle - $[25 - 50]$, and bottom - $[40 - 50]$.

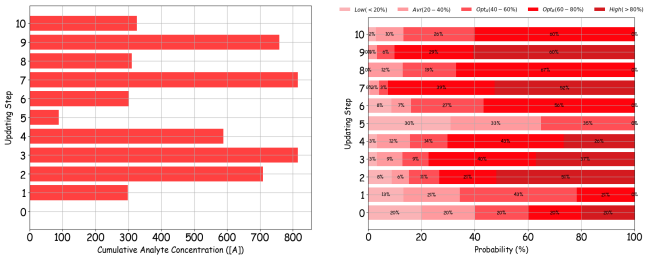


(a) (b) (c)

Fig. 7: Color tokens corresponding to the level of [A]; concentration of A (a) and the corresponding color tokens (b), and accuracy of detecting RCs for forty farms (c).

system and the centralized approach. Figure 7c shows that the proposed system achieves higher accuracy than the centralized approach. That is because the centralized approach independently decides the RC of each farm without taking into account the prior chemical usage at each step. On the other hand, the BC-IoNT uses information in the previous block for creating the current block to represent the chemical concentration status.

Moreover, stakeholders in the supply chain such as policy makers and government bodies may be interested in looking at the overall status of the level of chemicals rather than individual farms. That is because it could enable them to make an overall image of chemical usage and to generate alarming alerts. It will also help to identify whether there is any impact from external factors such as weather on varying chemical levels in the soil, even though the farmers may claim that a proper amount of chemical has been used. Therefore, to get an overall view on the level of chemicals over the area covered by the BC network, the change in the color tokens over time was explored. Figure8, for instance, depicts the color tokens obtained after ten chemical applications, including the corresponding cumulative sum of the chemical used in all forty farms. Similar to Figure 7, the probability of the overall status of the chemical level being in the RCs D and E is higher with the higher cumulative value of $[A]$.



(a) (b)

Fig. 8: Color tokens corresponding to overall variability in $[A]$, (a) level of A and (b) color tokens.

5 Discussion

In this section, based on the credit transactions between farmers and the miners, traceability of the use of chemicals is first discussed. Secondly, the credibility of each farm is explored to interpret how well farms are being compliant with chemical standards in the production process, followed by the advantages of the proposed system.

5.1 Traceability of the BC system

The proposed system can be used for ensuring traceability in farm produce. The traceability problems in this case are as (1) traceability in the amount of chemicals detected in a single farm, (2) traceability in the total amount of chemicals used on a farm, and (3) authenticity in the chemicals used on a single farm authentic, and is approved by the regulators.

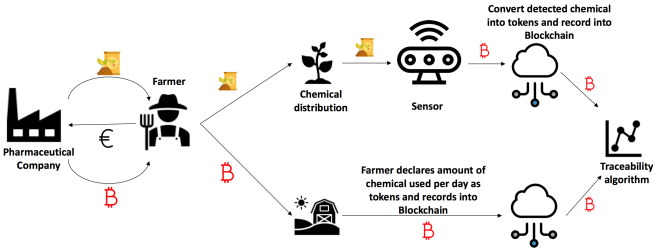
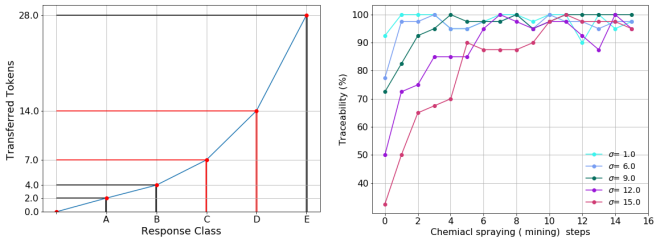


Fig. 9: Traceability assessment process.



(a) (b)

Fig. 10: Traceability evaluation, (a) amount of credits transferred corresponding to RCs and (b) traceability with the variability in $[A]$.

Our proposed BC-IoNT system can be used to solve these traceability problems as follows:

1. A farmer gets a fixed number of tokens with every purchase of chemicals from a company. These tokens can be given to the company, along with serial numbers to be used in the serialization of a standard unit of chemical. For example, for $1Kg$ bag of chemicals the company gets an unique serial number and a set of tokens (e.g., say 1000 tokens). After receiving the bags of chemicals and tokens in exchange for another fiat currency [4], leading to the farmer initiating two parallel traceability processes (Figure 9):
    a. In the first process, the farmer uses the chemicals as required in their farm. The sensor detects the levels of chemicals and creates a transaction from the farmer's account to the regulator's account to record the levels of chemicals detected.
    b. In the second process, the farmer decides the quantity of chemicals to be use in a day and makes a transaction from their account to a regulator's account with the amount corresponding to amount of chemicals that will be used. For example, if the farmer wants to use $1/10$ of a bag of chemicals, this will result in transfer of 100 tokens.
    c. A smart contract is executed by the regulators to check if the difference between the amount of tokens in these two processes is within a threshold. The threshold is calculated as per the expected accuracy of sensors.
2. This traceability approach can solve the above mentioned traceability problems:
    1. The level of chemicals on a farm can be traced by recording the area of farm from which the item is procured. This procedure will require labeling units of farms, associating each unit of farm with sensors and each farm produce with a set of nearby sensors or the unit of farmland.
    2. If the difference in the amount of tokens in both the traceability process is within a threshold, then it will certify that all chemicals used by the farm is traceable.
    3. If the difference in the amount of tokens in both traceability processes is within a threshold, then it will certify that all chemicals used by the farm is authentic. This is because the regulator supplies the tokens.

[4] a kind of national currency that has no intrinsic value and the value depends on the currency issuer such as country's central bank

Similar to the credit model explained in section 3.3.2, the amount of tokens transferred to the regulator (i.e., mining node) corresponding to the five RCs was exponentially increased as $e^{\alpha i}$, where $\alpha = 0.05$ and $i = 1, 2, \cdots, 5$ corresponds to the RCs $A, B \cdots, E$. That is because the concentration ranges corresponding to the RCs from A to E increases exponentially (Figure 2c). Figure 11 exhibits the amount of tokens transferred corresponding to the five RCs. By considering Figure 10a and 2c (i.e., RF with $[A]$) together, the amount of tokens required for different chemical levels can be decided effectively.

The amount of tokens that each farmer required to send to the regulator was computed based on the RC detected by using his color token. Also, each farmer computed the amounts of tokens based on the level of chemicals used over the farm. The difference between the amount of tokens from these two approaches was computed for forty farms for 15 chemical applications (i.e., mining steps). At each step, the number of farms that achieved the difference in the amount of tokens less than 10^-3 was computed. This token difference could, however, be affected due to the variability in $[A]$ (σ) and consequently, traceability can be varied substantially. Therefore, the impact of variability in $[A]$ on the traceability was also explored. Figure 10b exhibits the traceability obtained for four different values of σ. For each σ, the system achieved at least $\geq 98\%$ traceability though the number of mining steps required for achieving such a higher traceability is increasing with increasing σ value.

5.2 Credibility of farms

Assuming that initially every farm has 500 credits (i.e., $Cr_0 = 500$) and $\alpha = 0.05$, we explored the variability in credibility of each farm by computing the credits of each farm for $T = 1, \cdots, 100$. Figure 11 depicts the variability in the Cr for selected four farms out of forty farms with their frequency of being in the RCs D and E, respectively. The credits earning rate (i.e., steepness of the plots) is high when the frequency of being in the RCs D and E is less. This is because farms earn credits compared to the amount they spend when they are compliant with the chemical standards. Thus, the steepness of the credit curve is an indicator of the credibility. Besides, Figure 11 exhibits the variability in overall credits of forty farms. This helps to

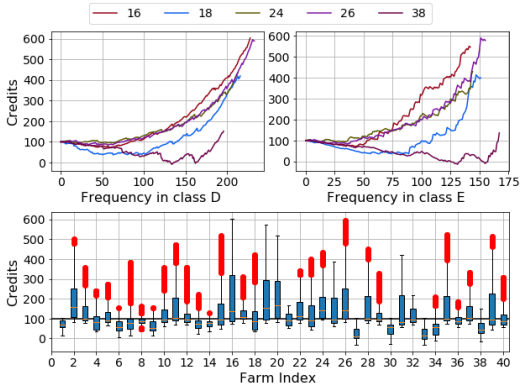


Fig. 11: Variability in credits for selected farms IDs (top) and overall credits (bottom).

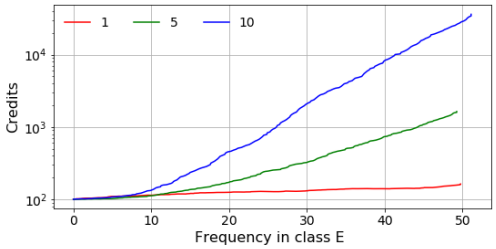


Fig. 12: Change of overall credits with tuning parameter α.

obtain a comparable idea about the variability in credibility of a set of farms and also to identify, for instance, the least credible farms easily, and they notify them to manage the use of chemicals.

The steepness could, however, be varied with the turning parametr α. To understand the impact of α, the amount of credits held by the set of forty farms was computed for three different values of α. Figure 12 exhibits the behavior of average credits of all forty farms with the average frequency of those farms being in the response class E. The steepness of the credit curve increases with increasing α. Thus, the credibility of farms can be changed by varying α.

5.3 Benefits of BC-IoNT

This system can be used for effective application of fertilizers. When applying different fertilizers based on their needs, variable-rate fertilization (VF) is one of the recommended and commonly used methods in Smart Farming. The VF method allows optimizing the soil available nutrient levels, ensuring sustainable productivity. It is critically important to have timely accurate information about the soil available nutrient contents in order to use the VF method, though factors such as spatial heterogeneity will limit obtaining this information effectively [22]. In this regard, the VF method combined with the BC-IoNT system will be a promising method to set up an effective fertilization strategy as the BC-IoNT system can be used to detect the soil nutrient content based on molecular-level data.

The proposed system can also contribute to improving the performance of the food supply chain. Consumers are more conscious about the quality and safety of food due to the lack of transparency in the food supply chain. As a viable solution, the color token generated in the BC-IoNT system can be use used as a reliable indicator (e.g., certificate) to represent the quality of food with respect to the use of chemicals in the food production process (e.g., identifying organic vegetables). At the same time, authorized parties, such as the FNs in the BC system, can assess the variability in the credibility of farms being compliant with chemical standards and send early warning alerts to producers to maintain an optimal chemical level. This could contribute to reducing farm input, for instance, expenditure on fertilizers and environmental impact. Therefore, the proposed approach has potential in empowering the validity of such essential features of the food supply.

6 Conclusion

This study focused on incorporating IoNT into BC systems to develop a distributed decision-making BC-IoNT for detecting the level of chemicals used in farms. The Langmuir molecular binding model and the Bayesian probability updating method based ML model was used as a smart contract in the system, which can effectively detect the levels of chemicals. This information can then be shared as a color token over the BC network. The data analysis confirmed that BC-IoNT system detected the level of chemicals in farms with higher accuracy than the centralized approach. The study concluded further that the efficiency of detecting the levels of chemicals could be varied due to several parameters such as variability in chemical concentration within as well as among farms. Based on this, our study found that selecting the optimal sampling frequency and the optimal number of probability updating steps is critical to improve the efficiency of the system in detection of the level of chemicals using nano-sensors. Moreover, the rate of change in the amount of credits held by each farm can be used as an indicator of the credibility to reward farms that are compliant with the recommended chemical standards. Similarly, the token-based traceability method confirmed that the BC-based system could achieve higher traceability, but the time taken for that could be varied with the variability in the level of chemicals over the farms.

This study is, however, a new research direction as the integration of IoNT into BC-enabled decision-making systems have not been considered before. Therefore, this systems can bring several advantages to several application domains and also lays the foundation for several future research directions as there are many challenges that need further attention.

Acknowledgment

This research was supported by the Science Foundation Ireland (SFI) project PrecisionDairy (ID: 13/1A/1977) as well as a research grant from Science Foundation Ireland and the Department of Agriculture, Food and Marine on behalf of the Government of Ireland under the Grant 16/RC/3835 (VistaMilk).

References

[1] M. Kuscu and O. B. Akan, "On the physical design of molecular communication receiver based on nanoscale biosensors," *IEEE Sensors*, vol. 16, no. 8, pp. 2228–2243, 2016.

[2] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "Blockchain challenges and opportunities: a survey," *IJWGS*, vol. 14, pp. 352–375, 2018.
[3] V. Sharma, I. You, F. Palmieri, D. N. K. Jayakody, and J. Li, "Secure and energy-efficient handover in fog networks using blockchain-based dmm," *IEEE Communications Magazine*, vol. 56, no. 5, pp. 22–30, 2018.
[4] H. Kim, J. Park, M. Bennis, and S.-L. Kim, "On-device federated learning via blockchain and its latency analysis," *CoRR*, vol. abs/1808.03949, 2018.
[5] M. S. Ali, M. Vecchio, M. Pincheira, K. Dolui, F. Antonelli, and M. H. Rehmani, "Applications of blockchains in the internet of things: A comprehensive survey," *IEEE Communication Surveys and Tutorials*, 2019.
[6] A. Dorri, S. S. Kanhere, , and R. Jurdak, "Blockchain in internet of things: Challenges and solutions," *arxiv.org*, 2016.
[7] G. Sylvester, *E-Agriculture in Action: Blockchain for Agriculture*. The Food and Agriculture Organization of the United States and the International Telecommunication Union, Bankok, 2019.
[8] X. Li, P. Jiang, T. Chen, X. Luo, and Q. Wen, "A survey on the security of blockchain systems," *arXiv*, 2018.
[9] J. Kang, R. Yu, X. Huang, S. Maharjan, Y. Zhang, and E. Hossain, "Enabling localized peer-to-peer electricity trading among plug-in hybrid electric vehicles using consortium blockchains," *IEEE Trans. on Industrial Informatics*, vol. 13, no. 6, pp. 3154–3164, 2017.
[10] K. Gai, Y. Wu, L. Zhu, L. Xu, and Y. Zhang, "Permissioned blockchain and edge computing empowered privacy-preserving smart grid networks," *IEEE Internet of Things*, 2019.
[11] M. A. Ferrag, M. Derdour, M. Mukherjee, A. Derhab, L. Maglaras, and H. Janicke, "Blockchain technologies for the internet of things: Research issues and challenges," *IEEE Internet of Things*, vol. 6, no. 2, pp. 2188–2204, 2019.
[12] Z. Yang, K. Yang, L. Lei, K. Zheng, and V. C. M. Leung, "Blockchain-based decentralized trust management in vehicular networks," *IEEE Internet of Things*, vol. 6, no. 2, pp. 1495–1505, 2019.
[13] S. Wang, J. Wang, X. Wang, T. Qiu, Y. Yuan, L. Ouyang, Y. Guo, and F.-Y. Wang, "Blockchain-powered parallel healthcare systems based on the acp approach," *IEEE Trans. on Computational Social Systems*, vol. 5, no. 4, pp. 942–950, 2018.
[14] Y. Sun, L. Zhang, G. Feng, B. Yang, B. Cao, and M. A. Imran, "Blockchain-enabled wireless internet of things: Performance analysis and optimal communication node deployment," *IEEE Internet of Things*, 2019.
[15] H. Shafagh, A. Hithnawi, L. Burkhalter, and S. Duquennoy, "Towards blockchain-based auditable storage and sharing of iot data," in *ACM Cloud Computing Security Workshop (CCSW17)*, Dallas, TX, USA, 2017.
[16] Y. Jiao, P. Wang, D. Niyato, and K. Suankaewmanee, "Auction mechanisms in cloud/fog computing resource allocation for public blockchain networks," *IEEE Trans. on Parallel and Distributed Systems*, 2019.
[17] C. Xu, K. Wang, P. Li, S. Guo, J. Luo, B. Ye, and M. Guo, "Making big data open in edges: A resource-efficient blockchain-based approach," *Sensors*, vol. 30, no. 4, pp. 870–882, 2019.
[18] S. Xu, J. Zhan, B. Man, S. Jiang, W. Yue, S. Gao, C. Guo, H. Liu, Z. Li, J. Wang, and Y. Zhou, "Real-time reliable determination of binding kinetics of dna hybridization using a multi-channel graphene biosensor," *Nature Communications*, vol. 8, 2017.
[19] P. Babington, *Biacore Assay Handbook*, aa ed. GE Healthcare Bio-Sciences AB, Bjorkgatan 30,75184 Uppsala, Sweden: GE Healthcare Life Sciences, 2012.
[20] X. Duan, Y. Li, N. K. Rajan, D. A. Routenberg, Y. Modis, and M. A. Reed, "Quantification of the affinities and kinetics of protein interactions using silicon nanowire biosensors," *Nature Nanotechnology*, vol. 7, p. 401407, 2012.
[21] B. Yin, L. Mei, Z. Jiang, and K. Wang, "Joint cloud collaboration mechanism between vehicle clouds based on blockchain," in *2019 IEEE International Conference on Service-Oriented System Engineering (SOSE)*, April 2019, pp. 227–2275.
[22] Z. Cheng, J. Meng, Y. Qiao, Y. Wang, W. Dong, and Y. Han, "Preliminary study of soil available nutrient simulation using a modified wofost model and time-series remote sensing observations," *Remote Sensing*, vol. 10, p. 64, 2018.

Dixon Vimalajeewa recieved his B.Sc in mathematics and statistics from The University of Ruhuna, Sri Lanka in 2012, and M.Sc in Computational Engineering from The Lappeenranta University of Technology, Finland in 2015. Currently, he is a PhD student at Telecommunications Software and Systems Group at Waterford Institute of Technology. His research interests include data analytics, sensor-based animal phenotypes and distributed learning algorithms.

Subhasis Thakur received his Ph.D from Griffith University, Australia in 2013. He has worked as research fellow at the University of Liverpool, the University of LAquila and the National University of Ireland. His research interest includes Blockchains, Multi-agent systems, Game theory and Cloud Computing.

John Breslin is a professor in Electronic Engineering and Director of TechInnovate at NUI Galway. He is also Co-Principal Investigator at the Insight Centre for Data Analytics at NUI Galway (formerly DERI), where he leads the Unit for Social Semantics. He created the SIOC framework, implemented in hundreds of applications on tens of thousands of websites. He has written 175 peer-reviewed publications and co-authored the books The Social Semantic Web and Social Semantic Web Mining. He is co-founder of boards.ie, adverts.ie, and the Galway City Innovation District / Porter-Shed. He is an advisor to AYLIEN, BuilderEngine and Pocket Anatomy. He has won various best paper awards (SEMANTICS, ICEGOV, ESWC, PELS) and two IIA Net Visionary Awards. He is Vice Chair of IFIPs Working Group 12.7 on Social Networking Semantics and Collective Intelligence. Dr Breslin is a Senior Member of the IEEE.

Donagh P Berry received his Bachelor in Agricultural Science and PhD in quantitative genetics at Unversity College Dublin, Ireland in 2000 and 2003, respectively and a Masters in Bioinformatics and Systems Biology from University College Cork in 2012. He is currently a senior principal investigator in quantitative geneticist at Teagasc, Ireland as well as being director of the VistaMilk Agri-Tech Research Centre. He holds professorships at three (inter)national universities. In his Teagasc capacity, he is responsible for the research on genetics in dairy cattle and is responsible for the development and implementation of genomic evaluations in dairy cattle, beef cattle and sheep in Ireland. As director of VistaMilk, he leads a team of > 200 scientists in the development and deployment of digital technologies in precision dairy production.

Sasitharan Balasubramaniam received the bachelors degree in electrical and electronic engineering from The University of Queensland in 1998, the masters degree in computer and communication engineering from the Queensland University of Technology in 1999, and the Ph.D. degree from The University of Queensland in 2005. He is currently an Academy of Finland Research Fellow at the Department of Electronic and Communication Engineering, Tampere University of Technology, Finland, and an Acting Director of Research at the Telecommunication Software and Systems Group, Waterford Institute of Technology, Ireland, where he was involved in a number of Science Foundation Ireland projects. His current research interests include molecular and nanocommunications, and Internet of (bionano) Things. He is on the Steering Committee of the ACM NanoCom Conference which he co-founded. In 2018, he received the ACM/IEEE NanoCom Outstanding Milestone Award, and he is also the IEEE Nanotechnology Council Distinguished Lecturer. He is currently an Editor of the IEEE INTERNET OF THINGS JOURNAL, Nano Communication Networks (Elsevier), and Digital Communication Networks.

Appendix F

Cooperative In-network Computation in Energy Harvesting Device Clouds

Journal Title:	Sustainable Computing: Informatics and Systems
Article Type	Refular Paper
Complete Author List	Chamil Kulatunga, Kriti Bhargava, Dixon Vimalajeewa, and Stepan Ivanov
Status	Published: vol. 16, pp. 106-116, Dec. 2017
Contribution	My contribution to this research article was identification of state-of-the-art in cooperative computing within WSN, recognizing limitations of existing techniques and modelling energy consumption of computation and communication tasks. I also took part in writing some parts of the paper and proof-reading.

Sustainable Computing: Informatics and Systems 16 (2017) 106–116

Contents lists available at ScienceDirect

Sustainable Computing: Informatics and Systems

journal homepage: www.elsevier.com/locate/suscom

Cooperative in-network computation in energy harvesting device clouds

Chamil Kulatunga*, Kriti Bhargava, Dixon Vimalajeewa, Stepan Ivanov

Telecommunications Software and Systems Group, Arclabs Research and Innovation Centre, Waterford Institute of Technology, Carriganore, Waterford, Ireland

ARTICLE INFO

Article history:
Received 10 July 2017
Accepted 9 October 2017
Available online 16 October 2017

Keywords:
In-network analytics
Cooperative computing
Computation offloading
Energy harvesting
Low-latency applications
Fog computing

ABSTRACT

The Internet of Things paradigm is creating an environment where the big data originators will be located at the edge of the Internet. Accordingly, data analytic infrastructure is also being relocated to the network edges, to fulfill the philosophy of data gravity, under the umbrella of Fog computing. The extreme edge of the hierarchical infrastructure consists of sensor devices that constitute the wireless sensor networks. The role of these devices has evolved tremendously over the past few years owing to significant improvements in their design and computational capabilities. Sensor devices, today, are not only capable of performing sense and send tasks but also certain kinds of in-network processing. As such, triple optimization of sensing, computing and communication tasks is required to facilitate the implementation of data analytics on the sensor devices. A sensor node may optimally partition a computation task, for instance, and offload sub-tasks to cooperative neighbouring nodes for parallel execution to, in turn, optimize the network resources. This approach is crucial, especially, for energy harvesting sensor devices where the energy profile and, therefore, the computation capability of each device differs depending on the node location and time of day. Accordingly, future in-network computing must capture the energy harvesting information of sensor nodes to jointly optimize the computation and communication within the network. In this paper, we present a theoretical model for computation offloading in micro-solar powered energy harvesting sensor devices. Optimum data partitioning to minimize the total energy consumption has been discussed based on the energy harvesting status of sensor nodes for different scenarios. The simulation results show that our model reduced both energy losses and waste due to energy conversion and overflows respectively compared to a data partitioning algorithm that offloads computation tasks without taking the energy harvesting status of nodes into consideration. Our approach also improves energy balance of a WSN which is an important factor for its long-term autonomous operation.

1. Introduction

With a growing number of devices in the Internet of Things (IoT) and high adopt-ability of cloud-based Big Data analytic platforms, the centralized cloud computing architecture has been recently challenged within the Internet community. Conventional cloud computing had been designed for monolithic applications assuming high availability of resources at large data centres. It saved CPEX for SMEs, particularly, the overall energy consumption of maintaining an Information and Communication Technologies (ICT) infrastructure. Furthermore, centralized clouds optimized resource utilization by statistically multiplexing peak-loads to avoid over-provisioning. This architecture functioned well until IoT devices generated some large datasets in remotely connected application domains such as smart agriculture [9] and Industry 4.0 [1]. Fog computing [26], is a new computing paradigm, that proposes the analysis of data (before aggregating it into big data sets) in a hierarchical and scalable way closer to the data sources. Although the term was coined by Cisco in 2012, the philosophy of data gravity where computation moves towards the data sources as far as they can, had been presented by Dave McCrory in 2010. Harnessing the computational power of the network devices for data processing has the potential to not only reduce the data in the backhaul network and, in turn, the latency experienced by the end users but also improve the overall energy consumption of the IoT platforms [10].

* Corresponding author.
E-mail addresses: ckulatunga@tssg.org (C. Kulatunga), kbhargava@tssg.org (K. Bhargava), dvimalajeewa@tssg.org (D. Vimalajeewa), sivanov@tssg.org (S. Ivanov).

https://doi.org/10.1016/j.suscom.2017.10.006
2210-5379/

This is particularly useful for applications in rural agriculture and Industry 4.0 where backhaul connectivity is limited between the remote rural farms/factories and the cloud [7].

A number of interpretations of Fog nodes have been proposed, to date. Authors in [2], for instance, discusses Mobile Edge Computing where mobile operators leverage resources of the edge devices in 5G rather than the centralized servers used in cloud computing for data processing. Several forms of ad-hoc cloudlets (micro-clouds) have been proposed in [4,18]. Certain studies have also extended the concept of Fog computing towards the extreme edge of the IoT in the private, enterprise, and community domains. This is primarily due to the design of pervasive low-power wireless technologies like ULP-PAN and LP-WAN as well as the tremendous improvement in computation capabilities of small devices (as mini-servers) such as CCTV cameras, mobile phones, and more recently, sensor devices that constitute Wireless Sensor Networks (WSN) [8]. In-network processing within WSN (referred here as in-network analytics) has been performed using different techniques such as data fusion, aggregation, compression and feature extraction [25,21].

It is of particular importance in latency-sensitive applications such as object tracking, intrusion detection, monitoring structural and machine failures, where the result of the processing may not be useful at all times, the response time at event detection is of the order of fraction of a second. As a result, while numerous studies in the past have focused on optimizing sensing and networking tasks to improve the energy efficiency of WSN, attention is being drawn towards triple optimization that includes on-board computation given the increased capabilities of sensor nodes. Maximizing computation within WSN through resource optimization is more desirable as future sensor nodes will be powered via energy harvesting, for continuous use, from background sources such as solar, wind, vibration and radio frequency [15].

Cooperative computing via computation offloading has been suggested for maximizing the use of in-network computational resources. In computation offloading, a device can select (sometimes in an opportunistic way [5,16]) a proximate infrastructure edge device (gateway) or another stationary or mobile device as an offloadee for parallel execution of tasks at different participating nodes [19]. Collaborative computing within WSN can enhance the capabilities of the resource constrained environment towards effective cyber-foraging approaches as shown in [20]. Multi-objective intelligent decisions can be made to optimize Fog computing resources and their application performance. The decision of how to optimally partition a task and where to offload given a completion time is an important research question which has not been much investigated in the literature. An analytical model for application partitioning in battery-powered WSN environment has been presented in [20]. An initiating node (IN) that is responsible for sensing data is designed that offloads partial computation to a neighbouring node known as the cooperating node (CN) such that the given task completion deadline is met while optimizing the energy resources of the network.

In this work, we consider in-network computation in WSN [14] and extend the cooperative computing approach discussed in [20] for different scenarios in an energy harvesting WSN. While in conventional WSN, the IN offloads less computation to CN owing to high communication energy, in case of energy-harvested nodes, the partitioning must be based on the level of stored energy as well as the current state of the device that determines the level of harvested energy. This is important to avoid over-flow of harvested energy (hence an energy waste) when battery is fully charged or energy conversion efficiency (75–65%) incurred by storing harvested energy into battery. Accordingly, we develop models for task partitioning to reduce the overall energy consumption of the network under different scenarios for latency-sensitive applications. Furthermore, we aim at improving the fairness within the network to ensure energy balancing. Our model and the simulation results show that our approach enables optimization of computation and communication for future energy harvested WSN and ensures sustainable operation.

2. Computational policies for clean energy

A node in a conventional sensor network forwards data without changing the payload. Instead, in-network processing allows a Fog node to not only function as a data source or merely relay a data chunk but also perform some computation on the data. In the early days of in-network processing, researchers were limited to a particular application within a sensor network such as calculation of average humidity or identifying a location of an event based on statistically correlated data aggregation. However, this is changing to embed more generic computational functionalities in WSN.

2.1. In-network cooperative computing in wireless sensor networks

In-network processing has been applied for data aggregation, fusion, compression and feature abstraction in WSN to save energy by reducing the number of bits and, in turn, data packets transmitted to a centralized server. Computations are performed at specific aggregation nodes (cluster heads) along the path to the destination node (gateway or server). Offloading decisions are, therefore, simple and based on the forwarding algorithm used such as LEACH to answer the question of where rather than what. This has progressed recently to use a swarm of heterogeneous nodes (such as sensors, actuators, robots, smart phones, drones, cameras) that collectively form an in-network analytic platform and requires specification of where as well as what to send. Authors in [11] propose for instance a new in-network computation algorithm based on channel fading to improve the reliability of aggregation function compared to simultaneously sending all or only one sensor reading.

Computation offloading is a useful distributed computing paradigm at different levels of network resources from large data centres to implanted nano-sensors. Highly available cloud computing provides VM/container level computing resources to the users to perform computation tasks in geographically distributed data centres. Mobile edge computing brings cloud resources into the edge of the operator-managed network to reduce core network traffic of the operator and provide low-latency for the users. Enterprise and community-cloud allow the installation of micro data centres that execute micro-services at the proximity of a company office or a community. The concept of cloudlets proposes the use of a set of mobile devices (different users) that collectively form an ad-hoc cloud [13]. Mobile computation offloading, for instance, can facilitate the execution of compute intensive tasks either on a nearby mobile (in terms of annotations) or on an infrastructure node (e.g. Androidx86).

Computation offloading in WSN is becoming increasingly important as the sensor devices exhibit improved capabilities in terms of computation power and reduced communication energy consumption. In conventional networks, sensor nodes transmit raw data to the sink node where some processing is performed and the results are communicated to the remote cloud. As a result, sensor nodes have prior knowledge of where and what to communicate. Moreover, the energy optimization is included in the algorithms. In modern-day WSN, sensor nodes can make on-the-fly decisions of where and what to compute under a subjected application completion deadline and, in turn, optimize energy usage. Therefore, the pre-designed computation offloading algorithms must be modified to make on-the-fly decisions. Accordingly, energy harvesting and

in-network processing can be combined to develop a sustainable and autonomous network operation.

2.2. Heterogeneity in energy harvesting sensor nodes

Computational sensor nodes, in future, will be powered using diverse natural energy harvesting sources such as solar, wind, radio-frequency, thermal, vibration or piezoelectric [22]. Such energy sources demonstrate random spatial–temporal generation patterns leading to heterogeneity in stored energy between sensor nodes in both outdoor and indoor environments. Changes in the temporal patterns might be significant only on a macro time scale. For instance, while weather may differ from one city to another on a single day at a given time, a sensor network on a smart farm will experience the same effect at the same time. On the contrary, spatial variations among co-located mobile sensor nodes may be obtained due to different orientations and obstacles, for e.g., presence of IMU and GPS modules [3] for animal mobility and location tracking under direct sunlight vs shadows. This heterogeneity will be higher, particularly, in outdoor WSN such as those used in agricultural practices for pasture-based dairy farming (e.g. laying animals with solar-covered tags), site-specific irrigation in cultivation (e.g. leaves may grow into or fall onto the senor nodes) and soil monitoring (e.g. shadows of the plants may cover the soil monitoring sensors).

Optimal energy management in such environments has been proposed using adaptive duty cycling, adaptive communication strategies, routing decision making and application policy management. Authors in [27], for instance, propose optimization of the duty cycle to maximize the common active time based on unpredictable heterogeneity of energy harvesting nodes. The authors propose both online and offline algorithms based on the probability of the harvested energy obtained using a real deployment environment.

We consider cooperation between such sensor nodes to collectively perform computation tasks under a heterogeneous energy harvesting environment. For example, each sensor node in such a scenario could partially perform some pre-processing or basic functional tasks such as averaging or compressing data. Balancing energy usage with computation offloading is important in such a Fog resource pooling environment due to three perspectives.

(a) Energy harvesting incurs a significant conversion loss while storing energy into a storage device like a battery or a capacitor. It accounts for about 25–35% of the total energy in battery storage and even higher for capacitors [27]. It is, therefore, preferable to use harvested energy directly whenever possible so as to minimize the conversion losses. Accordingly, any computation offloading to a node which is currently on solar power has a safe margin to use some energy to compensate for the communication overheads.

(b) If the amount of harvested energy is low, the system cannot perform both the charging and direct energy use operations together. That is, when the amount of harvested energy (E) is below a threshold (θ) a node must decide to either store the energy or use it directly but not both. Usually, in such situations, the most appropriate action is to store the harvested energy and consume the required energy from the battery. Therefore balancing stored energy within the nodes of a WSN is highly advantageous.

(c) Rechargeable batteries are a costly unit for energy harvesting sensor nodes. Therefore, they may have some limited capacity. Cooperative computing between the sensors is critical in such networks to optimize the energy usage via load balancing and avoid overflow of energy on nodes that are fully-charged with no computation task or energy deficit for others. Therefore, balancing energy consumption without using high capacity batteries is a positive trend in future WSN using energy harvesting.

2.3. Related work

Mobile computation offloading has been widely researched in the recent years with varied objectives such as energy saving, transparent code migration and scalability. An optimal technique for application partitioning and fair node selection between two homogeneous nodes has been discussed in [24]. Computation offloading in WSN, however, did not gain much attention until Sheng et al. [20] proposed optimal application partition and cooperation between two nodes to minimize overall energy consumption. Their work is based on cooperation between battery-powered homogeneous sensor nodes and assumes no selfish node behaviour. A cooperating node selection strategy that balances trade-off between fairness and energy consumption has been discussed.

Meanwhile, energy harvesting sensor nodes are becoming widely deployed and several studies discuss the heterogeneity in harvesting energy [15]. Dang et al. [6] presents predictive solar energy models for spatial–temporal weather conditions. Authors in [27] propose a stochastic duty cycling approach to minimize energy consumption by taking into account the heterogeneous energy harvesting sensor networks. In [25], authors discuss the importance of triple optimization of sensing, networking and in-network data processing based on energy harvesting. The authors have implemented an optimization algorithm to recycle wasted energy due to battery overflow in an energy harvesting WSN. In this paper, we extend the work done by [20] and propose an approach to balance the energy in computational sensor network using cooperative computing in energy harvesting networks. We apply this approach for the scenario where certain solar powered sensor nodes are under sunlight while others are obstructed by shadows for a certain duration within a day.

3. Modelling for cooperative computation

In this section, we present our application model, computation and communication energy consumption models, and the micro-solar based energy harvesting model.

3.1. Application model

In this work, we consider a lightweight analytic application that consists of a set of independent processing tasks to be computed cooperatively between two peer sensor nodes. We use the canonical model used in [28] to capture the essential characteristics of such a task-oriented application. Such tasks are normally arranged in a computational work-flow using a Dynamic Acyclic Graph (DAG) to be scheduled for execution in a distributed computing environment. A single processing task (A) is modelled with input data size (D) and a deadline for application completion (T). The Initiating Node (IN), which may be responsible for sensing the data, divides a single task into two sub-tasks for partial offloading to a target remote peer, referred to as the Cooperating Node (CN). The amount of processing data at the local node is denoted by L and the amount of data that is offloaded to the CN is denoted as R, where $D = L + R$. We assume there are no dependencies between the sub-tasks. For instance, in case of calculating average for a sensing variable, L and R may consist of n_L and n_R samples respectively. Note that, only R amount of input data is offloaded to the CN with no extra amount of code. We also assume that the response or the outcome of the processing sub-task at each node is negligible or locally consumed by another process. In the mentioned average calculation example, the local node will transmit only two values, which is the local average (A_L) and n_L, while CN will transmit its own local average (A_R) and n_R. An aggregation or the destination node will then calculate the overall average using the two responses from IN and CN.

3.2. Computation energy model

The energy consumption in embedded processors is dominated by dynamic power and can be regulated by the clock frequency using dynamic voltage and frequency scaling (DVFS) technique. Several attempts have been made to develop a simple and general computation energy estimation model for mobile and embedded processors. According to the literature, the computational energy consumption is proportional to the CPU load of a processor i.e. the number of CPU cycles required. Most of the work, therefore, considers the trade-off between energy (E) and task completion time (T) such that $E.T^{\alpha}$ is a constant for some values of α. In [24], the energy consumption for computing a task locally is calculated using Eq. (1), where K (in the order of 10^{-11} starting from ARM to Intel) is called the computation coefficient. The value of K depends on the effective switched capacity (determined by the chip architecture and the clock-frequency), the processing capability of the node, and the application completion probability used in the model in [28]. As evident in Eq. (1), a node consumes more energy for short completion deadlines T. A sensor node may, therefore, prefer more delay-tolerant tasks for local computation and offload tasks with large L and small T to a peer sensor node.

$$E_C = \frac{KL^3}{T^2} \tag{1}$$

3.3. Communication energy model

When a task is offloaded to another node, the energy used for communication depends on the number of bits transmitted [17]. This is energy consumed by the electronics in the physical layer and depends on the state of nodes – idle, transmit and receive. According to IEEE 802.15.4, energy consumption in the idle state can be neglected and, therefore, total energy consumption depends on the transmission of the number of bits at the sender and the reception of the same bits at the receiver which are equal in value but belong to two different nodes. A task can be scheduled for transmission to another node within one or more time-slots. This scheduling has been modelled using the Markov process based on whether the Additive White Gaussian Model (AWGN) channel state is good or bad. The energy used to communicate b bits within a time-slot t to another computational node depends on the path condition and the distance between the two nodes (represented as channel gain g) and is given by the following equation.

$$e = \frac{(2^b - 1)}{g}$$

According to one-shot channel allocation policy to transmit data task within a single time-slot, the scheduler must send L bits within one time-slot T. This is the simplest case in which all the data is sent within a single time-slot of communication window and the energy consumed is represented by a convex-monomial function as shown in Eq. (2).

$$E_t = \rho \frac{L^n}{g} \tag{2}$$

Here ρ is the communication coefficient of the link between the offloader and the offloaded and $g[0 \cdots 1]$ is the channel gain of the link that is calculated proportional to $1/d^2$ according to AWGN in free-space propagation where d is the distance between the two nodes. According to [20], transmission in one-shot policy ($n = 1$) only depends on the channel state and it is the most optimal approach for latency-sensitive applications. It also minimizes the time shift between local and remote computation since it assumes a negligible delay in over-the-air transmission. Moreover, it saves energy that is otherwise incurred by overhead scheduling due to data split across multiple time-slots.

3.4. Total energy requirement calculation per task

The total energy consumption owing to computation and communication during processing a task between two nodes can be calculated as the summation of four components as shown in Eq. (3). In [20], authors present the energy consumption for different input data sizes from 512 to 2048 bits. Here the job completion deadline is set to 20 ms, $K = 5 \times 10^{-11}$ and $\rho = 0.05$. For large data sizes, the gain in energy consumption is much better in case of using cooperative computing and varies with the values of the computation and communication coefficients. After a distance of 5 m, however, cooperative computing is not effective and localized computation becomes the preferred mode for the entire task according to their analysis.

$$\begin{aligned} \text{Total Energy}(E) = {} & \text{IN}\{\text{Computation} \quad L + \text{Transmission} \quad R\} \\ & + \text{CN}\{\text{Reception} \quad R + \text{Computation} \quad R\} \end{aligned} \tag{3}$$

In this paper we extend this approach by taking into account the energy harvesting state of the IN and CN nodes and also the energy conversion efficiency. We estimate the required equivalent energy (E) (i.e. before the conversion) from the energy harvesting source within the optimization algorithms.

3.5. Micro-solar energy harvesting model

We selected a latitude of 52° and longitude of −8° where the experimental smart farm for the project is located in Moorepark, Co. Cork, Ireland. We chose April 1st as the representative date of neither a winter day nor a summer day for the solar energy harvesting model. We model the solar energy harvesting pattern as a Gaussian curve (Fig. 2) with 8 h (T) clear sunlight from 8.00 am to 4.00 pm according to astronomical model developed by [23,12]. We consider a discrete time model with a time-slot of 1 min. A solar energy density of 15 mW cm^{-3} is assumed for 5 cm × 3 cm area on a micro-solar panel associated with a sensor node. This implies 735 μJ energy can be generated by a sensor node on a day without any clouds and obstacles shadowing it. We also modeled a shadow of 4 h which will randomly cover sensor nodes within the field. Micro-solar panel inclination was set to 90° and orientation to 45° in our model.

4. Energy-aware task partitioning

The aim of this work is to find the optimal data size for a task that is suitable for local computation (L) and remote computation (R) based on the state of harvested energy (under shadow, under sunlight with energy stored being under-flown and under sunlight with energy stored being over-flown) on both IN and CN. While we discuss the energy-aware application partitioning by IN and CN selection (in the following section), the energy state interchanges among the nodes using a distributed or a centralized approach is beyond the scope of this paper. The Lagrange Multiplier is used to solve the equal constrained optimization problem with an objective to minimize total required energy (E) from the solar panel at both nodes. When a task is to be processed at any given time, IN and CN may be in different states as shown in Table 1, resulting into different E_L and E_R values compared to non-energy harvesting-aware partitioning approach proposed by [20]. We calculate E accordingly as the summation of E_L and E_R values. We consider an energy gain factor λ as the reciprocal of the energy conversion efficiency in the equations for simplicity of deriving equations. For instance, $\lambda = 1.54$ represents 65% efficiency (Fig. 1) and implies that if a task consumes 10 μJ stored energy from the battery when the node is under a shadow, the value of E will be 20 μJ.

Table 1
Different energy harvesting states at IN and CN and the amount of total required energy in μJ using our energy-harvesting-aware task partitioning at $T = 20ms$. Local computing respectively consumes and computes 41.3 μJ (1024 bit).

			Initiating Node (IN)		
			Shadow	Light	
				Underflow	Overflow
Cooperating Node (CN)	Shadow		12.1 (526)	9.8 (469)	0 (1024)
	Light	Underflow	9.8 (472)	7.8 (526)	0 (1024)
		Overflow	1.6 (122)	1.1 (122)	0 (1024)

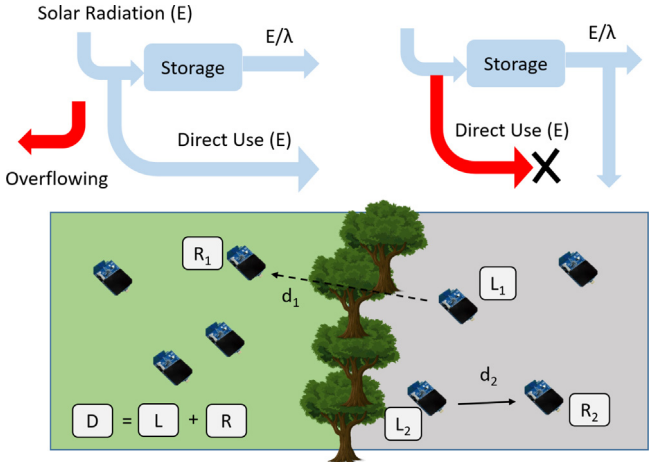


Fig. 1. Heterogeneity of energy harvested will be captured by an appropriate data partitioning and in-network computation offloading.

$$cos\theta = cos\alpha_p.cos\alpha_S + sin\alpha_p.sin\alpha_s.\cos(\beta_p - \beta_s)$$

$$cos\alpha_s = sin\delta.sinL + cos\delta.cosL.cosh \qquad sin\beta_s = -cos\delta.sinh/sin\alpha_s$$

$$x = 2\pi n/365, \qquad h = 15(t - 12)$$

$$\delta = 0.302 - 22.93cosx - o.229cos2x + 0.243cos3x + 3.851sinx + 0.002sin2x - 0.055sin3x$$

Fig. 2. The set of used energy harvesting astronomical modelling equations.

In the following sub-sections, we discuss the optimal task partitioning in terms of number of bits and the total energy required at both IN and CN to execute the task in μJ under the different IN and CN states (Table 1). The data size (D) is set to 1024 bits and the task completion deadline is changed from 5 to 100 ms. The channel gain between IN and CN is set to 0.9 and the values of K and ρ are 10^{-11} and 10^{-3} respectively. Energy gain factor $\lambda = 1.54$.

4.1. Shadow–shadow

When IN and CN are under shadow, both nodes consume energy from the stored battery power for task processing. Such a scenario does not incur any waste from the harvested energy. In this case, E can be calculated as the sum of local energy requirement E_L at IN (for computation of local task L and transmission of data R to CN) and remote energy requirement E_R at CN (for reception of data R from IN and computation of data R).

$$E = E_L + E_R = \{\alpha L^3 + \beta R\}\lambda + \{\beta R + \alpha R^3\}\lambda \tag{4}$$

On solving Eq. (4) using Lagrange constraint optimization in order to minimize E subjected to the constraint $L + R = D$, we obtain the values for L and R.

$$L = \frac{D}{2} + \frac{\beta}{3\alpha D} \text{ and } R = \frac{D}{2} - \frac{\beta}{3\alpha D}$$

Even though the amount of task partition is the same as in the non-energy harvesting case, the energy requirement is multiplied by the energy gain factor λ when we calculate the amount of surplus energy to be stored at each node. Fig. 3 shows that cooperative computing gains with low energy and the amount of the locally computed data increase with the task completion deadline. After a certain time of completion deadline, however, IN processes all the data locally and does not achieve any advantage by cooperating with a CN.

4.2. Shadow–light

In this case, the CN is under sunlight while energy is being harvested during the task processing. Therefore, remote computation R tends to be larger than in the previous case since energy required at the CN can be consumed directly from the energy harvesting source without incurring any conversion loss, if the battery is underflow (not charged up to the full capacity). Furthermore, it can use abundant energy if the battery overflows (battery fully charged and harvesting energy being wasted). Accordingly, we analyze this case separately for the two scenarios as the amount of L and R will be different.

Energy under-flowing: In this scenario, the energy is directly used from the solar panel at CN through the input regulator without incurring battery conversion loss. However, any surplus harvested energy can be stored in the CN battery without contributing towards energy waste as the battery is not charged to the full capacity. Therefore, E can be calculated as follows.

$$E = E_L + E_R = \{\alpha L^3 + \beta R\}\lambda + \{\beta R + \alpha R^3\} \tag{5}$$

On solving Eq. (5) to minimize E, we obtain the values for L and R as given below, where the value of A is obtained by solving the quadratic equation $aA^2 + bA + c = 0$ (see Appendix A) such that $L < D$.

$$L = \sqrt{\frac{A}{3\alpha\lambda}} \text{ and } R = \sqrt{\frac{A - (1+\lambda)\beta}{3\alpha)}}$$

Furthermore, values of a, b and c are calculated as follows.

$$a = (1 - \lambda)^2$$

$$b = 2\lambda(1 + \lambda)\{(1 - \lambda)\beta - 3\alpha D^2\}$$

$$c = \lambda^2[9\alpha^2 D^4 + \beta(1 + \lambda)\{6\alpha D^2 + (1 + \lambda)\beta\}]$$

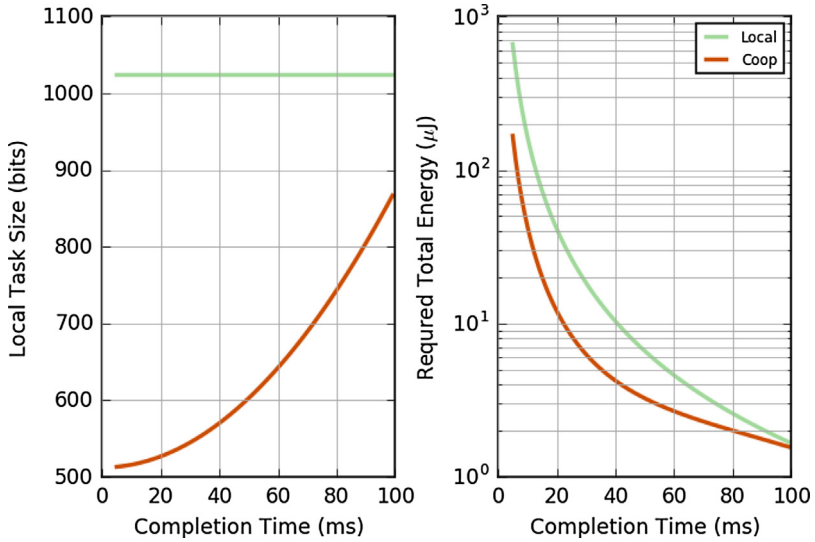


Fig. 3. Cooperative computing gains with low energy when both nodes are under shadows. However, it does not gain any energy saving when completion deadline is larger than 100 ms.

Energy over-flowing: If the battery at CN is fully charged, the energy required at CN is not considered for the total energy requirement calculation since CN in this case is wasting the harvested energy. However, transmission energy used for offloading data R to CN should be considered in the energy consumed at IN, which prevents offloading all the data D to CN.

$$E = E_L + E_R = \{\alpha L^3 + \beta R\}\lambda + \{0\} \tag{6}$$

On solving Eq. (6), we obtain $L = \sqrt{(\beta/3\alpha)}$ which is a trade-off between the required computation and communication energy at IN, and $R = D - L$. This shows that even though harvested energy at CN is wasted, IN cannot offload all the task to CN unless the completion deadline is very low.

As illustrated in Fig. 4, IN offloads more data to the CN when CN is under sunlight. We can see that if CN is overflowing, more computation can be offloaded than in the case of CN under-flowing. In case of the former, significant energy gain is observed for lower task completion deadlines when compared to the local computation only.

4.3. Light–shadow

When IN is under sunlight, the size of local computation L tends to be larger than in the previous case. This is because energy consumed at the IN can be used directly from the energy harvesting source without incurring conversion loss or from the energy being wasted according to the level of charge of the battery (similar to the previous case). Therefore, this case is also investigated under two scenarios where the amount of L and R is different.

Energy under-flowing: In this scenario, energy is directly used without conversion loss but harvested energy can be stored in the IN battery rather than being wasted. Therefore, E can be calculated as follows.

$$E = E_L + E_R = \{\alpha L^3 + \beta R\} + \{\beta R + \alpha R^3\}\lambda \tag{7}$$

On solving the optimization problem, we obtain the values for L and R as under.

$$L = \sqrt{\frac{A}{3\alpha}} \text{ and } R = \sqrt{\frac{A - (1+\lambda)\beta}{3\alpha\lambda}}$$

The value of A can be obtained by solving the quadratic equation $aA^2 + bA + c = 0$ such that $L < D$ using the following values of a, b and c.

$$a = (1-\lambda)^2$$
$$b = (1+\lambda)\{(\lambda-1)2\beta - 6\alpha\lambda D^2\}$$
$$c = 9\alpha^2\lambda^2 D^4 + (1+\lambda)\beta\{6\alpha\lambda D^2 + (1+\lambda)\beta\}$$

Energy over-flowing: In this scenario, the energy required at IN is not considered for the total required energy calculation since the node is wasting the harvested energy. Furthermore, all the computation is done locally at IN rather than offloading partial computation to CN. Accordingly, $E = E_L + E_R = 0 + 0$ and we obtain $L = D$ and $R = 0$. Fig. 5 shows that cooperative computing gains when IN is under sunlight.

4.4. Light–light

This case results in three possibilities for deciding the values of L and R. The calculation of the total required energy for each scenario is explained below.

Both nodes energy under-flowing: When both IN and CN are under sunlight without energy over-flowing, nodes can consume energy directly from the energy source and store surplus energy in the battery without any waste. In this case, E can be calculated as shown in Eq. (8), and the values of L and R can be calculated as in the shadow-shadow scenario in Section 4.1 (however the energy required at each node will be differed by a factor of λ).

$$E = E_L + E_R = \{\alpha L^3 + \beta R\} + \{\beta R + \alpha R^3\} \tag{8}$$

On solving Eq. (8) to minimize E subject to the condition $L + R = D$, we can obtain the values for L and R as under.

$$L = \frac{D}{2} + \frac{\beta}{3\alpha D} \text{ and } R = \frac{D}{2} - \frac{\beta}{3\alpha D}$$

IN energy over-flowing: In this scenario, all the processing takes place locally at the IN irrespective of the CN state and the energy required at IN is not considered for the total energy calcu-

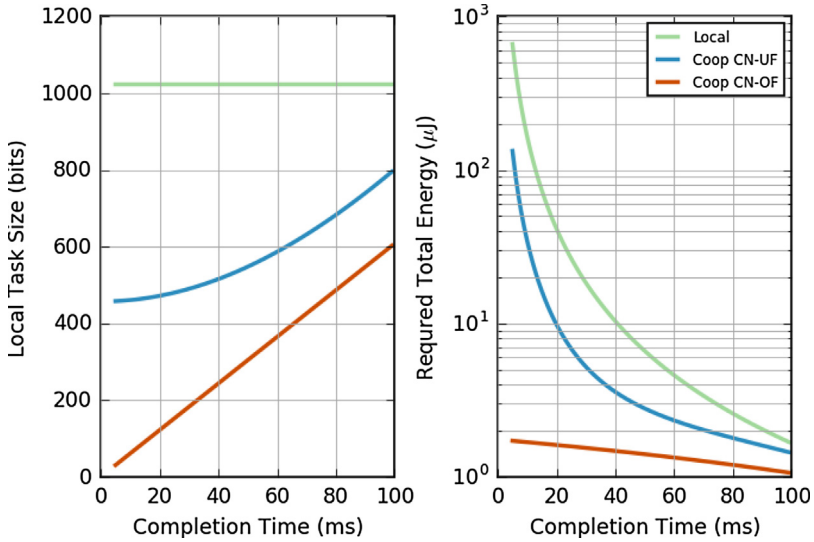


Fig. 4. IN offloads more data to CN when it is under sunlight. CN overflowing can achieve much lesser total energy consumption than underflowing scenario.

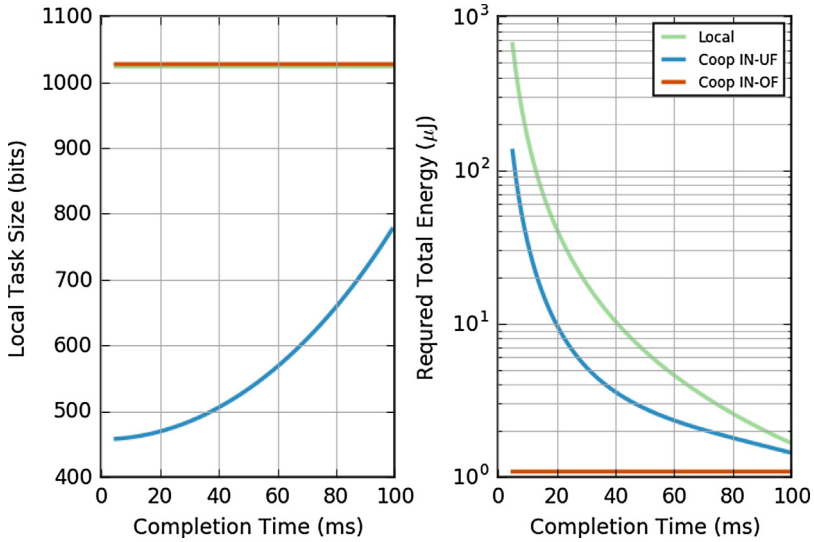


Fig. 5. Overflowing IN does not offload any data to a CN. However, underflowing IN offloads data in cooperative computing.

lation. Therefore, total energy is calculated as $E = E_L + E_R = 0 + 0$ and we obtain the $L = D$ and $R = 0$.

IN under-flowing and CN over-flowing: If the battery at CN is fully charged, the energy required at the CN is not considered for the total energy (E) calculation since CN, in this scenario, will waste the harvested energy. However, energy used for offloading data R to CN must be considered as the energy consumed at IN. Accordingly, total energy is calculated as given in Eq. (9).

$$E = E_L + E_R = \{\alpha L^3 + \beta R\} + \{\theta\} \tag{9}$$

We then obtain the value of $L = \sqrt{\beta/3\alpha}$ which is a trade-off between the required computation and communication energy at IN, and $R = D - L$. This shows that again even though harvested energy at CN is wasted, IN cannot offload all the data to CN. Also, the energy required by IN does not incur any conversion loss. Fig. 6 shows the gain in cooperative computing in these scenarios. As in the previous case, IN does not offload any data to CN in case of energy over-flowing whereas it offloads a considerable amount of data to CN when CN is over-flowing.

5. Energy-aware node selection strategy

The CN selection strategy must also be modified to make it suitable for our application model compared to non-energy harvesting scenario. In case of a non-energy harvesting environment, the minimum total energy strategy (MES), where the CN with minimum total cooperative energy cost is selected among the set of neighbouring nodes. This strategy does not consider past energy consumption (i.e. utilization). The drawback of it is that some nodes are overused due to cooperation and may lead to reduced battery lifetimes or many dead batteries, which affect the long-term autonomous functioning of the WSN. For example, a node in close proximity to a computationally-intensive node may cooperate heavily and may, therefore, be overused unfairly than what they save from cooperative computing.

In this work, CN selection is performed based on utility function as in [20], where authors define a utility function (U) based on the energy saved from cooperative computing compared to executing the complete data task locally at an IN. Our simplified utility function incorporating with the energy gain factor is given as follows.

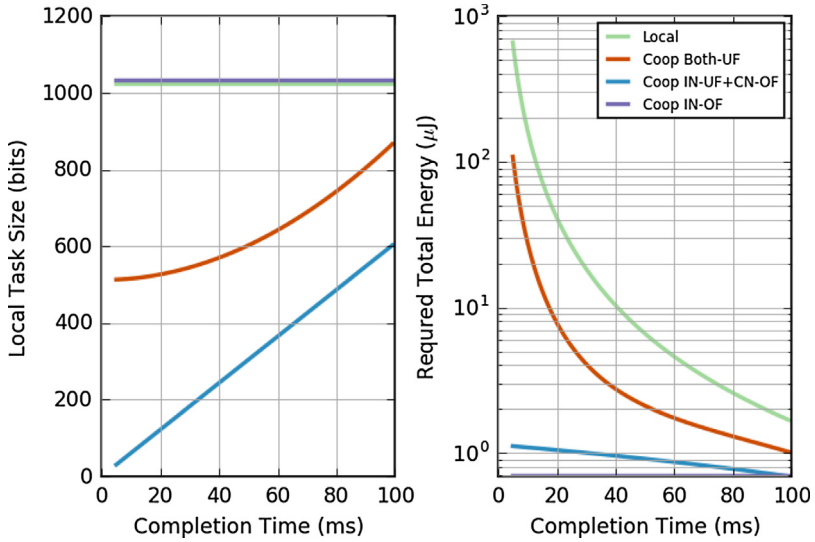


Fig. 6. IN does not offload any data when it is overflowing. However, IN does not offload all the data when CN is overflowing due to communication energy used at the IN.

$$U = \begin{cases} E_{LO} - E_L & \text{if IN} \\ -\gamma E_R & \text{if CN} \end{cases}$$

Here $E_{LO} = \frac{KD^3}{T^2}$ and $\gamma = 1$ if the CN is under sunlight and $\gamma = \lambda$ if CN is under shadow and under-flowing. The value of $\gamma = 0$ if the CN is over-flowing energy. Utility of IN will not change as the impact of the sunlight is already calculated in the required energy optimization. A Cooperation Index (CI) is then defined based on the cumulative utility as given below for $t = 0$ to $t - 1$ same as in [20]. A node can be used as a CN at time t if and only if the value of CI is positive.

$$CI = \begin{cases} 1 & \text{if } U(0 : t-1) \geq 0 \\ 0 & \text{if } U(0 : t-1) < 0 \end{cases}$$

This strategy is called positive utility strategy (PUS) [20]. Larger utility will have a higher chance to be selected as a CN. Designing an algorithm for this process based on the harvested energy (either in the past or predicted) is beyond the scope of this paper and remains as our future work.

6. Performance evaluation

We simulated our energy harvesting-aware computation offloading algorithm (e-COFF) with 30 energy harvesting sensor nodes using the *SimGrid* simulator.[1] Nodes were randomly located within a 10 m × 10 m geographical space. We selected latitude of 53°, where the project site is located and day of the year as 91 (01st April) in the micro-solar energy harvesting model, which harvested energy in a sinusoidal pattern within a day. We used randomly distributed obstacles for shadowing for a duration of 4 h. The size of the solar panel at a node was selected as 5 cm × 3 cm, which determined the multiplication factor of the sinusoidal harvesting pattern. We update the stored and wasted energy at each node per minute based on the harvested and the consumed energy during that period. We compared our results with non-energy harvesting-aware data offloading algorithm (COFF).

A computational task was created every 2 s randomly by a selected senor node in the WSN with a size (D) of 1024 bits. We

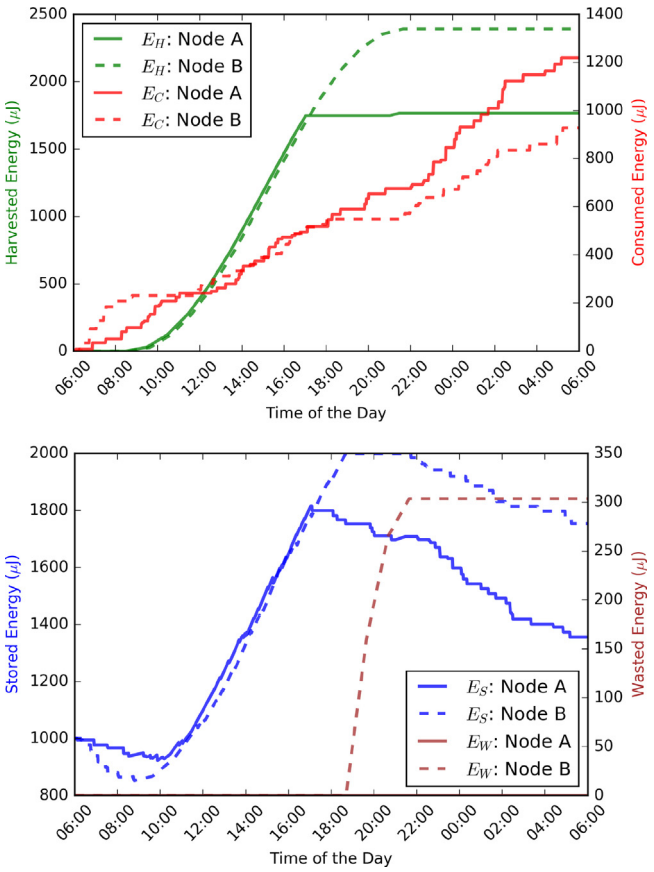


Fig. 7. The amounts of measured energy performance parameters at two different energy harvesting sensor nodes for duration of 24 h. Full battery capacity = 2000 mAh.

used a maximum capacity for a battery storage of a sensor node as 2000 mAh and set it to its half at the start of the day. Harvested (E_H), required to consume (E_C), stored (E_S) and wasted (E_W) energy at the end of 24 h duration from 6.00 am were measured. Task completion time (T), harvesting energy gain factor (λ), K and ρ were set respectively as 20 ms, 1.54, 10^{-11} and 0.001 unless otherwise changed in some sections. We calculated channel gain (g) according to the

[1] http://simgrid.gforge.inria.fr.

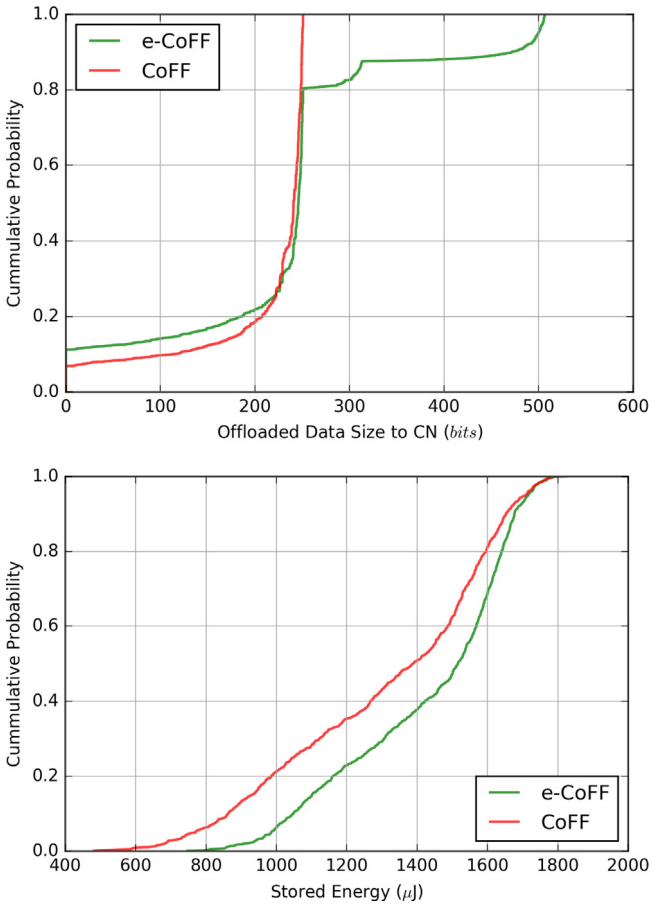


Fig. 8. Top: CDF of the sizes of data chunks being offloaded to a remote CN (R). Bottom: CDF of the stored energy (E_S) at the end of the day. K = 10^{-11}, ρ = 0.001, D = 1024 bits, T = 20 ms.

free-space wave propagation of AWGN as,

$$g = \frac{1}{\sigma\sqrt{2\pi}}\exp^{-d^2/2\sigma^2}$$

where we selected σ as 8 in our simulations and d was calculated in the units of m.

As shown in Fig. 7, the harvested energy (E_H) of Node B does not experience any shadow while Node A experiences shadow during the day. Moreover, Node A demands slightly more energy (i.e. required energy (E_C) for task executions either as an IN or CN before being converted) than Node B. As we can see in the bottom graph, Node B saturated with stored energy (E_S) from 6.00 pm to 8.00 pm resulting in a waste of energy (E_W). Node A's battery capacity does not overflow at any given time and therefore does not experience any waste of energy. This validates our chosen relative values of energy performance parameters in order to fulfill a requirement of self-sustainability of the wireless sensor network.

We then observe the probability distribution of the offloaded task sizes to a CN (R) and the end of the day stored energy (E_S) for the two algorithms; e-COFF and COFF. The top and the bottom graphs of Fig. 8 shows the cumulative probability densities of R and (E_S) respectively with 30 different seed values set in the simulator. As we can see e-COFF offloads more data to a CN than the COFF algorithm does. The second figure shows COFF leaves with more sensor nodes towards lower energy levels at the end of the day while e-COFF leaves more stored energy towards higher energy levels.

Fig. 9 shows the difference between the consumed energy of COFF and e-COFF (E_C of COFF $- E_C$ of e-COFF). We have changed

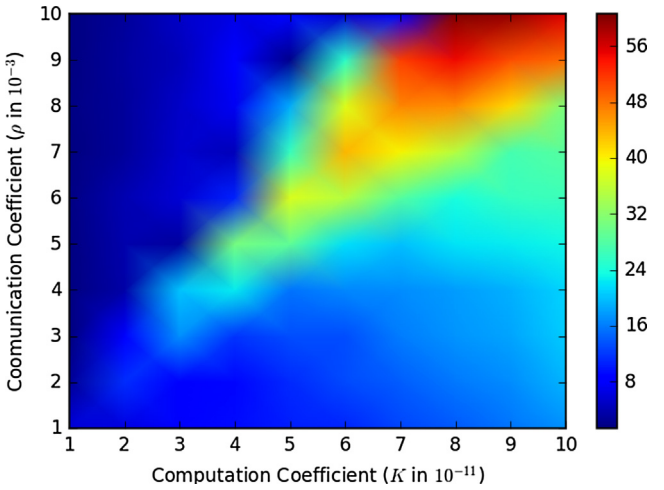


Fig. 9. The difference of consumed energy (E_C) in mJ between the energy-unaware (COFF) and our energy-aware (e-COFF) data partitioning and computation offloading algorithms (task completion deadlines = 20 ms). Both used Positive Utility Strategy (PUS) in selecting a CN.

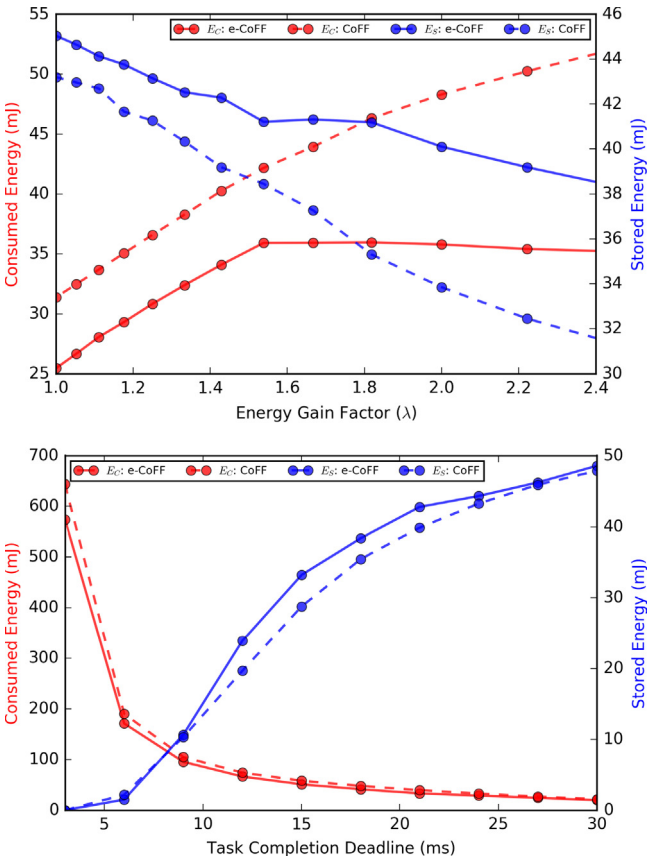


Fig. 10. Consumed and stored energy of the two algorithms for different energy gain factors (λ) when t = 20 ms (top) and for different task completion deadlines when λ = 1.54 (bottom).

the computation coefficients (K) in the range of 10^{-11}–10^{-10} and the communication coefficient (ρ) in the range of 0.01 and 0.001 both with a step size of 0.1. According to the figure, the performance improvements of the e-COFF is apparent for all the values of computation and communication coefficients since all the values in figure are positive. When both K and ρ are higher (top-left corner), performance improvement is significant.

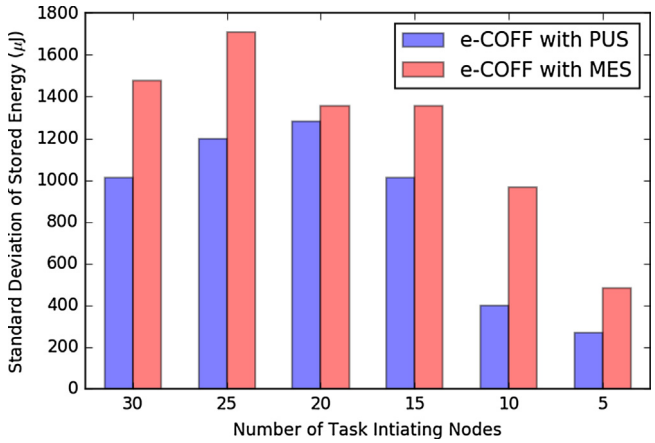


Fig. 11. The standard deviation (STD) of the stored energy (E_S), where a smaller STD indicates a better energy balance, of 30 micro-solar energy harvesting sensor nodes at the end of the day (completion deadline = 20 ms and $\lambda = 1.54$).

Next, we change (top graph in 10) the energy gain factor (λ) from 2.5 to 1.0 (i.e. energy conversion efficiency from 0.4 to 1.0) with a step of 0.5 while keeping T at 20 ms. In another experiment we also change task completion deadline (bottom graph in 10) from 5 ms to 30 ms with a step size of 5 ms while keeping λ at 1.54. Figures show the consumed energy during the day and the stored energy at the end of the day. According to the figure at the top, e-COFF shows lesser (E_C) than the COFF. Our algorithm also shows that stored energy performance is also higher compared to COFF. Performance improvement of e-COFF is much better when energy gain factor λ is low. However, the performance improvement is not very apparent for the changing range of task completion deadline (T).

We then localize the task generations only to a subset of sensor nodes to investigate the adverse impact of the overuse of energy at a CN. In this case, we reduced the number of task originating nodes from 30 (all, which is the same as before) to 5 with a step size of 5. We used two CN selection strategies; MES and PUS, with our e-COFF algorithm. Fig. 11 shows the standard deviation of the end of the day stored energy E_S, which is lower with the PUS strategy. It shows that the impact using the utilization factor in micro-energy harvesting where, if energy level of a node is low, becoming a CN persistently is critical. According to the figure use of CI solves the problem of overuse of CNs by INs in a computation intensive hotspots.

7. Conclusions

Energy-aware cooperative computing is a key technology that will benefit from energy harvesting in Fog computing applications. It is particularly important when the energy harvesting patterns and obstructions are dynamic, thereby, creating spatially heterogeneous energy sources. In this paper, we extend the optimal data partitioning algorithms developed for computation offloading by taking into account the state of the energy being harvested at the heterogeneous nodes. We evaluate our e-COFF algorithm under different scenarios and compare with COFF algorithm. Our results illustrate that overall energy consumption can be improved in a WSN by minimizing energy losses due to a poor energy conversion efficiency and waste due to energy overflows under constrained energy storage capacities. Our algorithm preformed the optimized data portioning with a positive utility cooperating node selection strategy, which balances the stored energy of the sensor nodes at the end of a day, which is useful concern for the sustainability of a WSN using micro-scale energy harvesting sources.

Acknowledgements

This work has received support from the Science Foundation Ireland (SFI) and the Agriculture and Food Development Authority, Ireland (Teagasc) as part of the SFI-Teagasc Future Agri-Food Partnership, in a project [13/IA/1977] titled “Using precision technologies, technology platforms and computational biology to increase the economic and environmental sustainability of pasture based production systems”.

Appendix A

In this appendix, we discuss the optimal data partitioning for a scenario where the Initiating Node (IN) is under shadow while the Cooperating Node (CN) is under sunlight. The energy required by IN is obtained directly from the harvested energy whereas the energy required by CN is obtained from the battery. The total energy consumed is calculated as follows.

$$E = E_L + E_R = \{\alpha L^3 + \beta R\} + \{\beta R + \alpha R^3\} + A\{D - L - R\}$$

Using gradient optimization with partial derivatives, we get

$$\frac{\partial E}{\partial L} = 3\alpha L^2 - A \rightarrow L^2 = \frac{A}{3\alpha}$$

$$\frac{\partial E}{\partial R} = \beta + \lambda\beta + 3\alpha R^2 - A \rightarrow R^2 = \frac{A - (1+\lambda)\beta}{3\alpha}$$

After solving the equation $(L+R)^2 = D^2$, we get a quadratic equation $aA^2 + bA + c = 0$ to find the roots for A.

References

[1] E. Baccarelli, P. Naranjo, M. Scarpiniti, M. Shojafar, J. Abawajy, Fog of everything: energy-efficient networked computing architectures, research challenges, and a case study, IEEE Access 5 (2017) 9882–9910.
[2] S. Barbarossa, S. Sardellitti, P.D. Lorenzo, Communicating while computing: distributed mobile cloud computing over 5G heterogeneous networks, IEEE Signal Process. Mag. 31 (6) (2014) 45–55.
[3] K. Bhargava, S. Ivanov, C. Kulatunga, W. Donnelly, Fog-enabled WSN system for animal behavior analysis in precision dairy, IEEE International Conference on Computing, Networking and Communications (ICNC) (2017).
[4] M. Chen, Y. Hao, Y. Li, C.F. Lai, D. Wu, On the computation offloading at ad hoc cloudlet: architecture and service modes, IEEE Commun. Mag. 53 (6) (2015) 18–24.
[5] M. Conti, M. Kumar, Opportunities in opportunistic computing, IEEE Comput. Mag. 43 (1) (2010).
[6] N. Dang, E. Bozorgzadeh, N. Venkatasubramanian, QuARES: a quality-aware renewable energy-driven sensing framework, Sustain. Comput.: Inf. Syst. 2 (4) (2012) 171–183.
[7] M. Eto, R. Katsuma, M. Tamai, K. Yasumoto, Efficient coverage of agricultural field with mobile sensors by predicting solar power generation, 2015 IEEE 29th International Conference on Advanced Information Networking and Applications (2015) 62–69.
[8] A. Giridhar, P.R. Kumar, Toward a theory of in-network computation in wireless sensor networks, IEEE Commun. Mag. 44 (4) (2006) 98–107.
[9] A. Grogan, Smart farming, IET Eng. Technol. Mag. 7 (6) (2012).
[10] F.J.K. Hinton, R. Ayre, T. Alpcan, R.S. Tucker, Fog computing may help to save energy in cloud computing, IEEE J. Sel. Areas Commun. 34 (5) (2016) 1728–1739.
[11] S.-W. Jeon, B.C. Jung, Opportunistic function computation for wireless sensor networks, IEEE Trans. Wirel. Commun. 15 (6) (2016).
[12] J. Jeong, D. Culler, A practical theory of micro-solar power sensor networks, ACM Trans. Sens. Netw. 9 (1) (2012) 1–36.
[13] S. Jeong, O. Simeone, J. Kang, Mobile cloud computing with a UAV-mounted cloudlet: optimal bit allocation for communication and computation, IET Commun. 11 (7) (2017) 969–974.
[14] I. Khan, F. Belqasmi, R. Glitho, N. Crespi, M. Morrow, P. Polakos, Wireless sensor network virtualization: early architecture and research perspectives, IEEE Netw. 29 (3) (2015) 104–112.
[15] M.L. Ku, Y. Chen, K.J.R. Liu, Data-driven stochastic models and policies for energy harvesting sensor communications, IEEE J. Sel. Areas Commun. 33 (8) (2015) 1505–1520.
[16] C. Kulatunga, L. Shalloo, W. Donnelly, E. Robson, S. Ivanov, Opportunistic wireless networking for smart dairy farming, IEEE IT Prof. Mag. 19 (2) (2017) 16–23.

[17] J. Lee, N. Jindal, Energy-efficient scheduling of delay constrained traffic over fading channels, IEEE Trans. Wirel. Commun. 8 (4) (2009) 1866–1875.
[18] G.A. Lewis, S. Echeverría, S. Simanta, B. Bradshaw, J. Root, Cloudlet-based cyber-foraging for mobile systems in resource-constrained edge environments, Companion Proceedings of the 36th International Conference on Software Engineering (2014).
[19] A. Mtibaa, K.A. Harras, K. Habak, M. Ammar, E. Zegura, Towards mobile opportunistic computing, IEEE Conference of Cloud Computing (2015).
[20] Z. Sheng, C. Mahapatra, V. Leung, M. Chen, P. Sahu, Energy efficient cooperative computing in mobile wireless sensor networks, IEEE Trans. Cloud Comput. (2015) 1.
[21] I. Stojmenovic, Machine-to-machine communications with in-network data aggregation, processing, and actuation for large-scale cyber-physical systems, IEEE Internet Things J. 1 (2) (2014) 122–128.
[22] S. Sudevalayam, P. Kulkarni, Energy harvesting sensor nodes: survey and implications, IEEE Commun. Surv. Tutor. 13 (3) (2011) 443–461.
[23] J. Taneja, J. Jeong, D. Culler, Design, Modeling, and Capacity Planning for Micro-Solar Power Sensor Network, 2008.
[24] L.M. Vaquero, L. Rodero-Merino, Finding your way in the fog: towards a comprehensive definition of fog computing, ACM SIGCOMM Comput. Commun. Rev. 44 (25) (2014).
[25] S. Yang, Y. Tahir, P.Y. Chen, A. Marshall, J. McCann, Distributed optimization in energy harvesting sensor networks with dynamic in-network data processing, IEEE INFOCOM 2016 – The 35th Annual IEEE International Conference on Computer Communications (2016) 1–9.
[26] B. Zhang, N. Mor, J. Kolb, D.S. Chan, K. Lutz, E. Allman, J. Wawrzynek, E. Lee, J. Kubiatowicz, The cloud is not enough: saving IoT from the cloud, 7th USENIX Workshop on Hot Topics in Cloud Computing (HotCloud 15) (2015) 12–21.
[27] J. Zhang, M. Wang, Z. Li, Stochastic duty cycling for heterogeneous energy harvesting networks, 2015 IEEE 34th International Performance Computing and Communications Conference (IPCCC) (2015) 1–9.
[28] W. Zhang, Y. Wen, K. Guan, D. Kilper, H. Luo, D.O. Wu, Energy-optimal mobile cloud computing under stochastic wireless channel, IEEE Trans. Wirel. Commun. 12 (9) (2013) 4569–4581.

Chamil Kulatunga received BSc in Electronics and Telecommunication Engineering degree from University of Moratuwa, Sri Lanka in 1999, MSc in Telecommunications from Waterford Institute of Technology (WIT), Ireland in 2003 and PhD in Internet Engineering from University of Aberdeen, UK in 2009. He is currently working as a post-doctoral researcher in Telecommunications Software and Systems Group (TSSG), WIT, Ireland. His research interests include Distributed Data Analytics, Fog Computing, Internet QoS and Smart Agriculture.

Kriti Bhargava received Bachelor's-Master's of Technology degree in Computer Science and Engineering from LNM Institute of Information Technology, Jaipur, India, in 2014. She is currently pursuing PhD in Science at TSSG, WIT, Ireland. Her research interests include Internet of Things, Fog Computing and Data Mining.

Dixon Vimalajeewa received Bachelor's of Science degree in Mathematics and Statistics from University of Ruhuna, Sri Lanka, in 2012 and Master's of Science degree in Computational Engineering from Lappeenranta University of Technology, Finland, in 2015. He is currently pursuing PhD in Science at TSSG, WIT, Ireland. His research interests include Distributed Machine Learning Algorithms, Data Analytics and Mathematical Modelling.

Stepan Ivanov received Diploma (with honours) in Applied Mathematics and Informatics from Moscow State University, Russia, in 2007 and PhD in Science from WIT, Ireland, in 2013. He is currently working as a post-doctoral researcher in TSSG, WIT, Ireland. His research interests include Wireless Communications, Internet of Things, Fog Computing and Edge Analytics.