ARCADA

Food Waste Prediction in Grocery Stores

Time Series Forecasting by Deep Learning

Weijing Shi

Master's Thesis
Master of Engineering - Big Data Analytics
May 16, 2022

MASTER'S THESIS	
Arcada University of Applied Sciences	
Degree Programme:	Master of Engineering - Big Data Analytics
Identification number:	8635
Author:	Weijing Shi
Title:	Food Waste Prediction in Grocery Stores Time Series Forecasting by Deep Learning
Supervisor (Arcada):	Amin Majd
Commissioned by:	

Abstract:
Food waste has becoming an increasingly important problem globally. Source of waste derives from supply chain, food manufacturing, household, retail stores etc. This thesis focuses on the food waste problem in retail industry and aiming to predict the potential food waste in a grocery store by using deep learning approaches. With a real world data-set from a grocery store in Finland, various deep learning models - MLP, CNN, LSTM, GRU have been trained to forecast the upcoming food waste on product level. The outcome of the experiments have been evaluated by means of calculating the RMSE value as well as a business oriented confusion matrix. The study has demonstrated the capability of the selected deep learning models on predicting the future food waste in retail context.

Keywords:	foodwaste, timeseries forecasting, deep learning
Number of pages:	44
Language:	English
Date of acceptance:	1.10.2054

CONTENTS

FIGURES

TABLES

ABBREVIATIONS

FAO	Food and Agriculture Organization of the United Nations
UNEP	UN Environment Programme
ML	Machine Learning
FNN	Feedforward Neural Networks
CNN	Convolutional Neural Networks
RNN	Recurrent Neural Networks
LSTM	Long Short Term Memory
DIF	Demand Influence Factor
IQR	Interquartile Range

FOREWORD

After over 12 years working experience in ERP consulting , I have decided to enhance my skill sets in the data science area and started the Big Data Analytic master degree studies in the middle of the Covid-19 pandemic. It is not easy to study a new field besides the full time work, the intensive study as well as the thesis project required and consumed abundant of time, focus, patience and determination. Thanks to the support of the teachers in Arcada, my former colleagues in the case company of the thesis , my family and also myself, finally I am finishing the writing of the thesis, and accomplishing a second master degree in my life.

I am about to start a new career journey in data and business consulting tomorrow. It just feels more than happy to write the Foreward part of the thesis at this moment.

Weijing Shi

Helsinki, 13.2.2022

1 INTRODUCTION

In the recent decades food wastage has becoming a rising public concern globally, in the context where climate change is constantly worsening. According to (FAO 2011), around 1/3 of the food in the world was estimated to be lost or wasted annually, and food waste alone generates about 8 to 10 percentage of global greenhouse gas emissions (UNEP 2021). As a result, the reduce of food waste can directly lead to the decrease the greenhouse emission and thus help slowing down the global warming. Generally speaking, food waste can be produced throughout the entire food value chain, mainly in the following areas: agricultural production, post-harvest handling and storage, raw food processing, distribution in wholesale and retail markets as well as individual household consumption (FAO 2011). This thesis focuses on the food waste problem in retail industry, specifically in the grocery stores. A real-life data-set from one of the biggest food stores in Finland has been studied. The data-set consists of past two and half years wastage history of about 1500 products in Ready-to-Eat Meals category, along with other relevant features which could impact the wastage e.g. the daily sales number, the stock situation, holidays and promotions etc. By means of trying various machine learning and deep learning models on the given data, the object is to examine how well ML methods would be able to predict the potential upcoming food waste for each product. The model trained in this study with satisfying prediction result can be taken into use by the case company as food waste prediction service in its production environment.

1.1 Background

The first chapter serves as background overview, where the overall food waste situation at retail level as well as at case company level are being introduced.

1.1.1 Food Waste at Retail Industry

According to Stenmarck (2016) retail sector is believed to produce about 5% of the total food waste in EU. Available data from the existing researches show that fruit and vegetables, dairy products, bread and fresh meat products are the most wasted products at retail level (Felicitas Schneider 2020). There are several common reasons for food being wasted or discarded in the grocery stores such as: expired shelf-life, package damages,

overstocking due to inaccurate demand prediction and so on. Even the wastage amount is relatively low compare to other players in the food supply chain i.e. food waste in household or production process, retailer can play an important role in reducing the waste because of its unique position in the value chain. First of all, retailer has the capability to sell the potential waste with their special pricing techniques, for instance the products approaching best before dates with over half price discount are usually highly attractive to many of the grocery consumers. Secondly retail giants with great procurement power are capable of setting high standard of goods and services from its manufacturers and logistics partners, which could reduce the potential food waste being generated prior goods arrive to stores due to bad logistic handling or manufacturing faults. Last but not the least, from retailer's own business perspective, aiming high in its operational excellence for example improving the demand forecast accuracy can benefit the overstocking situation, so that goods will not be ordered too much than the actual need and thus avoid being wasted.

In summary it can be concluded that retail industry does not produce as much food waste as other players in the food value chain, but retailers can effectively influence the food wastage situation with its unique role.

1.1.2 Food Waste in Case Company

In this section, the background of the case company together with its motivation and strategy of dealing with food waste problem is introduced. The case company who provides the data-set for this study is a leading Finnish grocery operator, who owns over 1200 food stores all over Finland with 1.2 million daily customer visits (GROCERY TRADE 2021) . The company is dedicated for sustainable development, and aims to be carbon neutral in 2025 and zero emissions by 2030 (anualreport 2021). Food waste is calculated as carbon emission thus reducing of which has been considered as one of the concrete action plans to be achieved in order to realize the sustainability goal. Based on the company's published annual reports, the company is making good progress in terms of food waste reduction. In year 2016, the identified food waste relative to sales is calculated to be 13% in the case company's grocery chains, and the figure has been further reduced to 12% by the end of 2020.

1.2 Aim of the project

The aim of this thesis is to experiment how well deep learning models such as Convolutional Neural Network and Recurrent Neural Network can help to forecast the potential upcoming food waste in a supermarket based on the historical data . It is believed that the waste prediction in good quality can help the store to plan the actions in advance on the products which are possibly being wasted, so as to largely reduced the waste being generated.

1.3 Research Question

In order to reach the aim of the project described above, the research question is formed as follows:

Predict the upcoming food waste in grocery store based on the historical transactional data.

The research question is a typical time series forecasting problem. Deep learning or artificial neural network are the main methods to tackle the problem in this study.

1.4 Limitations

The data-set is derived from one hypermarket in an industrialized country and the concerned merchandise category is ready-to-eat meal. Therefore the presented results is limited to similar type of environment and context.

1.5 Ethical considerations

This thesis does not concern any sensitive personal data.

2 LITERATURE REVIEW

Time series forecasting is used widely in many applications such as stock price forecasting, whether prediction, traffic forecasting and so on, in this study it is applied to the food waste problem in a retail grocery store. In the following sections, the relevant theories are reviewed, including machine learning, deep learning and their applications on the time series problems.

2.1 Machine Learning

Machine learning is a branch of Artificial Intelligence(AI) which imitates the way how humans learn (IBM 2021), to deal with the unseen data and future situation by creating the models and algorithms utilizing the historical data. The biggest difference of ML and transitional programming is that ML models do not have to be explicitly programmed (Samuel 2000). Through times of iterations the accuracy improves gradually accordingly. Two most famous machine learning algorithms are so called supervised learning and unsupervised learning.

Supervised Learning Supervised Learning refers to such situation, where the algorithms are trained under human's overseeing. The original data set consists of the tagged label along with the data features. For example an image itself contains the data features such as colors and shapes and its label can be a dog or a cat. By training a machine learning model with thousands of such labeled images, a machine learning model could learn how to classify dog or cat on a new image. Regression and Classification are the two most well-known and popular problems using supervised learning. In case of a regression problem the goal of the model is to predict the output in terms of numerical value e.g. predict how much a stock price will be, while classification model is aiming to predict a categorical value instead for instance to tell whether the stock price will go up or down tomorrow.

Unsupervised Learning Unsupervised Learning means the algorithms are not supervised by humans as the training data is not labeled. Clustering and Association Analysis are the common problems using unsupervised learning. The purpose of a Clustering prob-

lem is to group the data points into desired sizes without telling the model which specific conditions to follow for the grouping. A typical use case of clustering is to create the customer groups with similar shopping behavior based on the receipt data. Association Analysis on the other hand is meant for discovering the relations between the variables. In the context of grocery trade and huge amount of receipts as training data, associate analysis can help to find out what products are most often purchased by the consumers.

2.2 Time Series Forecasting

Time series can be defined as accumulating random variables in chronological order (Dinesh C.S. Bisht 2021) . Time series forecasting is the process of predicting the future events by analyzing the past happenings, assuming the past trend will continue in the future. The process includes modeling the historical data and fitting the model to the same set of variables to get the future values. A time series consists of base, trend, season and residual components. The base is long-term mean of the time series and the trend is long-term movement of the mean value. Seasonal behavior meant for the cyclically repeated changes and residuals are the stochastic components of a time-series data (Lars Kegel & Lehner 2018). These factors together structure the models applied in the time-series forecasting problems. In general, time series forecasting method involves two classes of algorithms, they are:

- Linear models
- Non Linear models

Linear models are traditional statistical models such as AR, MA, ARIMA and SES, while deep learning models are usually considered as non linear due to its widely usage of activation functions and they are the main methods to be experimented in this study.

2.3 Traditional statistical models

Classic time series models have a longer tradition and rooted in statistics and mathematics. They usually learn from past observations and therefore predict future values using solely recent history, such as Autoregressive Integrated Moving Average (ARIMA), and Simple

Exponential Smoothing (SES). ARIMA is one of the popular and widely adopted time series analysis methods, developed by Box and Jenkins (1976). ARIMA is derived from AR(autoregressive) and MA(moving average) method's, and is meant for fitting a class of linear time series models. ARIMA model is proper for stationary time series data, SES model is on the other hand appropriate for non-stationary data (i.e. data with a trend and seasonal data). The limitation of the linear models are however that they require variables to be independent with each other, and do not account for the latent dynamics existing in the data (Selvin et al. 2017).

2.4 Deep learning models

In the recent few decades, deep learning models have been seen great success and many research research papers have successfully applied deep learning methods. Deep Learning is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain called artificial neural networks, therefore deep learning models are often referred to as deep neural networks. A typical neural network architecture is illustrated in the Figure 1 (Bahi & Batouche 2018), which is consist of an input layer,one or more hidden layers and an output layer. In each layer there are several nodes, or neurons, and the nodes in each layer use the outputs of all nodes in the previous layer as inputs, so that all neurons interconnect with each other through the different layers. Each neuron typically is assigned a weight that is adjusted during the learning process and decreases or increases in the weight change the strength of that neuron's signal. The commonly used neural networks types include:

- Multilayer perceptrons (MLP)
- Convolutional neural network (CNN)
- Recurrent neural network (RNN)

Like traditional machine learning models, deep learning can work on supervised problem e.g. regression and classification, as well as unsupervised problem like clustering. In the later paragraphs, we are reviewing the algorithms of these common deep learning models, by means of which the food waste data-set has been experimented.

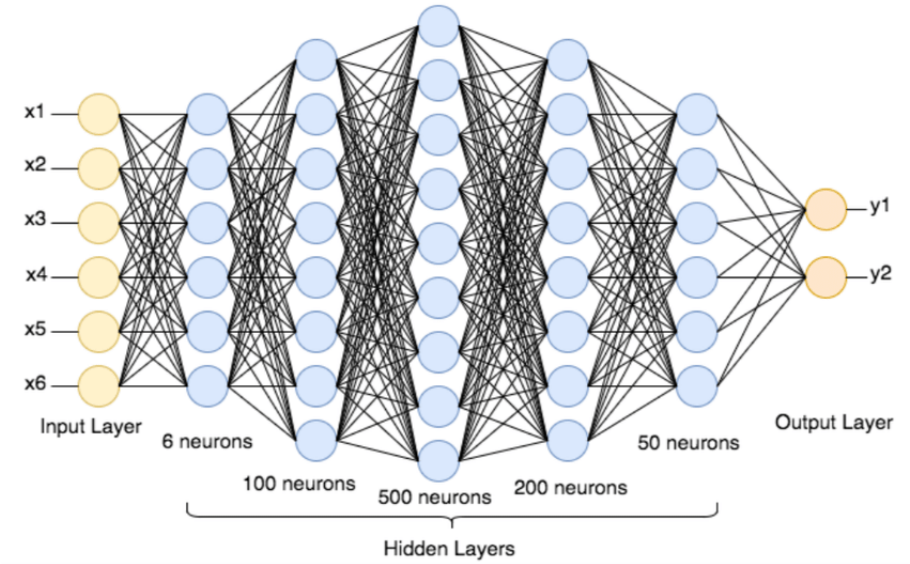

Figure 1. deep learning architecture
(Bahi & Batouche 2018)

2.4.1 Multilayer perceptrons MLP

Feedforward neural networks (FNN) allow signals to travel only in one direction, from input to output. There are no cycles or loops in the network thus it is considered as the simplest type of artificial neural netowork. Multilayer perceptrons(MLPs) is one special type of FNN, where nodes from one layer are connected (using interconnections or links) to all nodes in the adjacent layer(s) (Lek & Park 2008). Figure 1 is actually a MLP network. The major use cases of MLP are pattern classification, recognition, prediction and approximation (Abirami & Chitra 2020). MLP model can deal with non-linear problems like other deep learning models, however according to scikit learn documentation it has some disadvantages such as different validation accuracy per different random weight initialization, effort required of hyperparameters finetunning and sensitive to feature scaling. MLP is used as baseline model in this study.

2.4.2 Convolutional neural network CNN

CNN is another type of feedforward neural networks widely used in image recognition and text classification. It came to be known since late 80s and transformed the world of computer vision and audio processing due to its unique capability of encoding spatial relationships (Rivas 2020). A standard CNN architecture consists of several convolutional layers, pooling layers, as well as fully connected layers as shown in Figure 2. Convolution

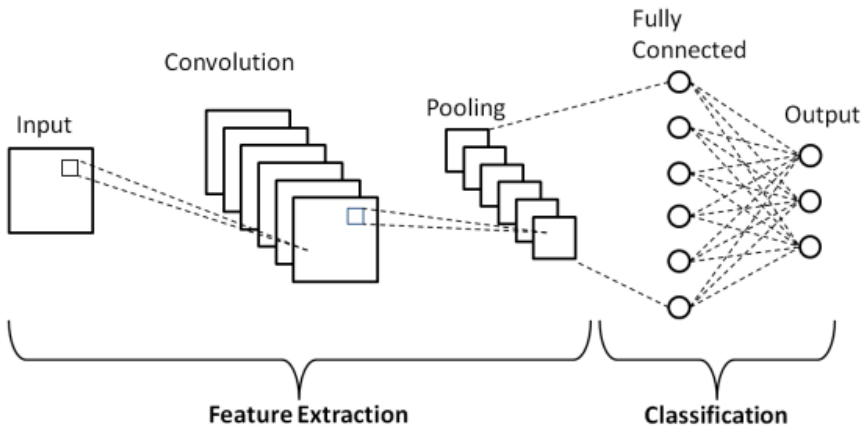

Figure 2. CNN architecture
(Blaji 2020)

is one type of matrix multiplication that are applied to the original input object or the previous set of feature maps, in order to capture the relevant features. Pooling layers are intended to reduce the number of computations by reducing the dimensionality of the problem, most popular ones are e.g. AveragePooling and MaxPooling. Fully connected layers are usually put before the classification output of a CNN and are used to flatten the results before classification. CNN can also be applied in the time series problem due to its ability of feature extraction, thus is also experimented in this project.

2.4.3 Recurrent neural network RNN

In contrast to feedforward neural networks, recurrent neural network RNN refers to such artificial neural network where loops exist within the hidden layers. Thanks to such setup RNN is able to use the information derived from previous step into the current task so that the network can understand the sequences better. RNN models is valuable in handling sequenced objects, thus is commonly applied in tasks such as speech recognition and language translation. The cost of the RNN though is the additional parameters and computations due to the weights associated with the input and previous output (Rivas 2020). In addtion, traditional RNN's in practise does not behave well in learning the long term dependencies. As a result some advanced RNN models e.g. LSTM, GRU are developed for improving the long-term memory. Figure 3 illustrates the architecture's of standard LSTM and GRU models. The details of each model are explained in the following paragraphs.

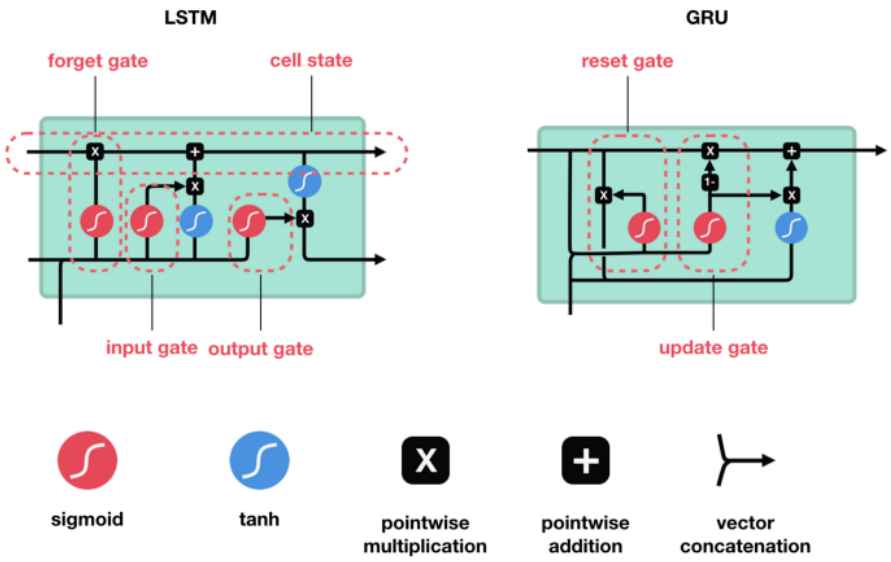

Figure 3. LSTM and GRU architecture
(Phi 2018)

LSTM Long Short Term Memory networks – usually just called "LSTMs" – are a special kind of RNN, capable of learning long-term dependencies. LSTM was introduced by S. Hochreiter (1997) and meant for addressing the problems with traditional RNN's, including vanishing gradients, exploding gradients and inability to remember or forget certain aspects of the input sequences (Rivas 2020). Three types of gates are the key components in LSTM networks which makes it different with traditional RNNs, they are so called forget gate, input gate and output gate as shown in Figure 2. The gates controls how information is flowing through the cells, and can learn what information should be kept or forgot during the process. These mechanism are trainable and optimized for each and every single dataset of sequences (Rivas 2020). Therefore LSTM is particularly suitable in dealing with sequenced data e.g. text, speech and general time-series data.

GRU Gated Recurrent Unit (GRU) is another type of newer version of RNN, aiming to improve the vanishing gradients problem associated with traditional RNN. The design of GRU is similar to LSTM, which contains two type of gates, update gate and reset gate. The update gate helps the model to determine how much of the past information to be passed to the future and the rest gate on the other hand decides how much of the past to be forgotten.

As explained, LSTM and GRU are both advanced versions of RNN and good in handling sequenced data-set, therefore both models have been experimented to the food waste data. In the later chapters, the experiments conducted in this study along with its result are shared.

3 METHODS

So far the previous chapters have clarified the business problem to be addressed and the theories behind the relevant deep learning models being experimented. In the Methods part, the research methodology of the study is checked. The process chart in Figure 4 is created to illustrate the key components of the applied methods and the logical relationship between each other. In the upcoming paragraphs, we start by introducing the data collection process, and continue with exploring the raw data to catch some general insights. Pre-processing activities are then being explained on how to get the data ready for feeding the selected deep learning models. Finally the evaluation approaches are described on how the experiment results have being measured.

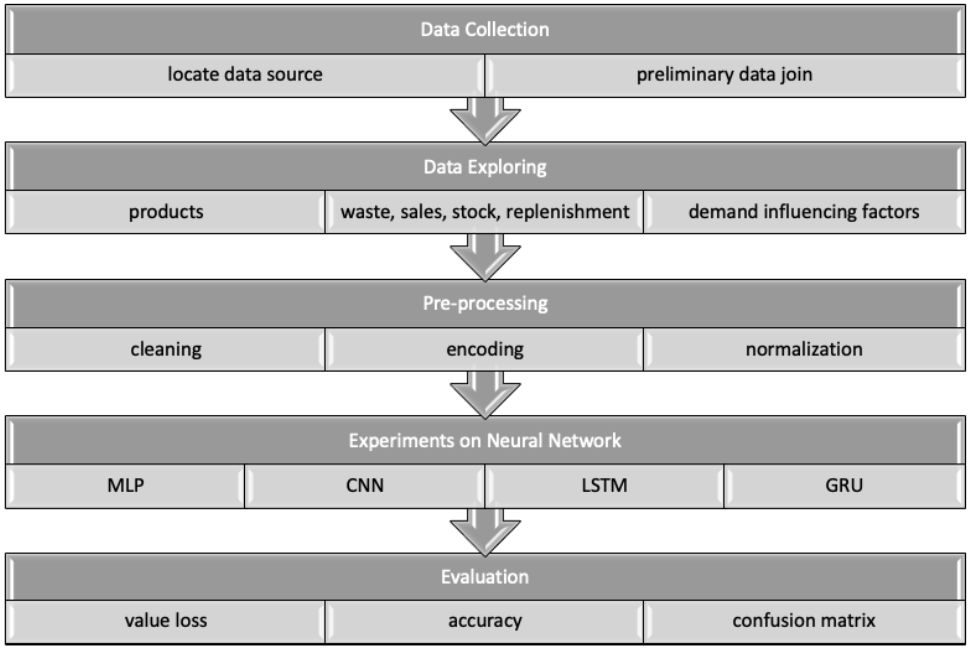

Figure 4. Research Methodology

3.1 Data

The data-set is prepared by the author from scratch in the purpose of this project. The scope of the data set is decided together with the domain expert in the case company as follows:

- Store Type: hypermarket

Figure 5. Ready-to-eat Meal, 2022.

- Product Group: ready-to-eat food
- Time Period: Jan 2019 - Sep 2021

The Store The case company has about 36.8% market share in Finnish food trade market in year 2021. The store selected for the study belongs to the hypermarket chain of the case company, which combines a department store and a grocery supermarket. In year 2020 there were 81 such stores all around Finland. The store in question is located in one of the most popular shopping center in the Helsinki Capital area. It offers a wide variety of assortment that can fulfill most of the households' daily consumption need.

Ready-to-eat Food The products included in the data-set belong to the ready-to-eat meals category. Such kind of food usually have been cooked or prepared in advance and can being eaten directly, for example the individual packed salad, soup and wok as shown in the Figure 5. Ready-to-eat food has some common characteristics i.e. easy

Table 1. Dataset Structure

Data Template								
Product	Date	Stock	Sales	Waste	GdsRceipt	DIF ID	DIF Grp	WkDay
10002000	01.01.2019	5	10	1	6	ABC	X0	2
10002000	02.01.2019	6	8	0	0	BCD	X1	3
...	...	...	...	...	...	...	...	...

to use, convenient package, storage in cool temperature, and relatively short shelf life, which lead to the fact that food in this category is easier to become waste compared to e.g. processed food like biscuit or tuna can.

Time Period The time series in question are between Jan 2019 and Sep 2021. As we known since 2020 spring when Covid-19 pandemic started, since then consumers' grocery shopping behavior has largely changed and also reflected in the demand of the readymade food. The data-set has collected days before and after the start of pandemic so that we shall be able to see how well the models are capable of dealing with such consumption change due to external demand influence factors.

Based on the scope of the data described above, the data collection process has started accordingly.

3.1.1 Data Collection

The data collection task begins with identifying the relevant features. The factors which might cause or effect the food waste in store are considered to be relevant and included to the data collection process. According to such principle the features are defined as follows: actual waste amount, stock balance, daily sales, incoming replenishment, and demand influence factors i.e. holidays, promotions and so on. The next step is to locate and extract these data from the Information Technology landscape of the case company. After rounds of the interviews and discussions with the experts in the relevant departments, most of the concerned data is found in the business data warehouse system and the forecast and replenishment system. It has then taken several days to download the data from different sources, and merged into one big CSV file in the format as shown in Table 1.

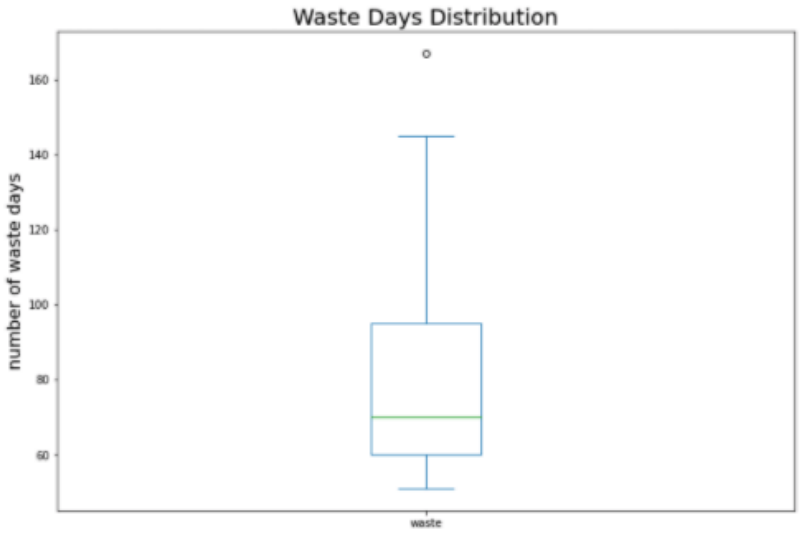

Figure 6. Waste Frequency

As can be seen from Table 1, the primary key of the data-set consists of product ID and the date, which means each row is meant for one particular product on one particular day. The first four features are in numerical values, they are the stock balance at the end of the day, and the total quantity sold or wasted or received on that day. The fifth and sixth features are related to Demand Influcence Factor (DIF). DIF ID refers to the identification of the DIF for example DAD is meant for Father's Day and MOM for Mother's day, while DIF Group combines similar type of DIF ID together. With the previous example both DAD and MOM are in same DIF Group e.g. H01. The last feature is week of the day, aiming to find the cyclical patterns in weekly basis. Next we will explore the content of the data-set for some general insight.

3.1.2 Data Exploring

There are initially over 1500 products in the raw data, which is compiled with the criteria: 1) target store, 2) target product group, 3) has consumption history since Jan 2019. Since time series prediction is in general under such assumption that the future activities would follow the similar way of working as of what has happened before, the deep learning models require such training data which has enough waste record. As a result the initial data-set has been further cleaned by filtering out the products without enough historical data, and 45 products eventually remain in the final data-set which have over 800 days valid sales and stock data, as well as at least 50 days positive waste history.

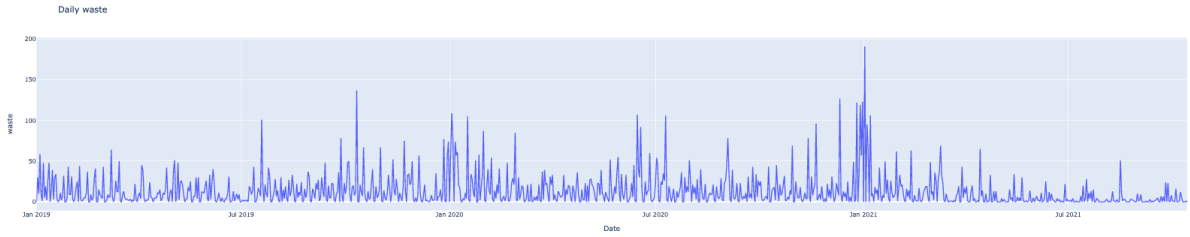

Figure 7. Daily Waste Overview

Unbalanced Data-set Let's first have a look how frequent waste is happening among these 45 ready-to-eat products. As can be seen from Figure 6, for most of the products, the number of total days with waste record is less than 3 months concerning the 33 months in study scope. The median value is around 70 days and the most frequently wasted product has waste record in 167 days during the study period. We can also conclude that the given data-set is imbalanced, since the number of days with waste is much less than without waste.

Cycle and Season Secondly we will check the cyclical or seasonal behavior of the given time series data. Figure 7 illustrates the daily aggregated waste quantity of all the products in question over the past years. A weekly cycle can be easily identified from the chart that Friday often reaches the peak of the food waste of that week and meanwhile Sunday is usually the troughs. This finding well reflects the labour shift schedule of the waste inspection and disposal activities in the store. There is though no obvious seasonal movement can be found in the past two and half years. Figure 8 provides another view by means of aggregating the numerical features on each day of the week from Monday to Sunday, which confirmed the previous finding that waste is normally happening on the working days. On the other hand, we can also see that sales and stock balance do not have obvious cyclical pattern, while the target store is usually receiving the replenishment of the concerned product group on Monday, Wednesday and Friday.

3.1.3 Data Pre-processing

Data pre-processing refers to the technique of preparing the raw data to make it suitable for feeding a Machine Learning models for the training purposes. The pre-processing approaches used in this study include: data cleaning, encoding and normalization.

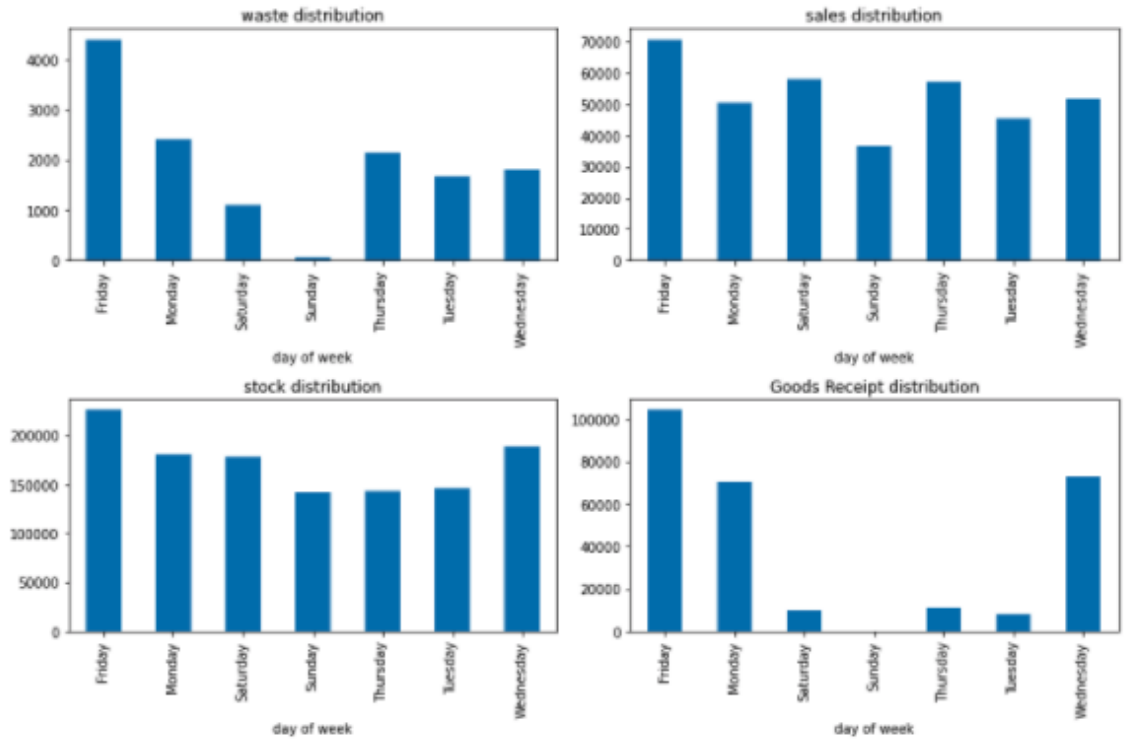

Figure 8. Day of the Week Distribution

Data Cleaning In the data cleaning step, the products without enough data points have been removed from the raw data. Here we define enough data in the way that 1) a product must have more than 800 days valid stock balance data, 2) a product must have more than 800 days valid sales data, and 3) a product must have more than 50 days waste record. The number of the products are thus being reduced from 1500 to 45.

In addition, the missing values have also been handled at this phase. The missing values are mostly found in the numerical features. According to the data source and the meaning of each feature, they have been processed so that missing sales and stock value would use the previous day's corresponding figure, and missing waste and replenishment would be having value zero.

Data Encoding In case of data encoding, the categorical features need to be converted to numerical values in order to be recognized by the ML models. Feature DIF ID and DIF Group are the categorical features to be converted in our data-set, and Pandas get_dummies method has been used for performing the encoding conversion. Figure 9 provides an overview on number of the features every product would have after encoding. Most prod-

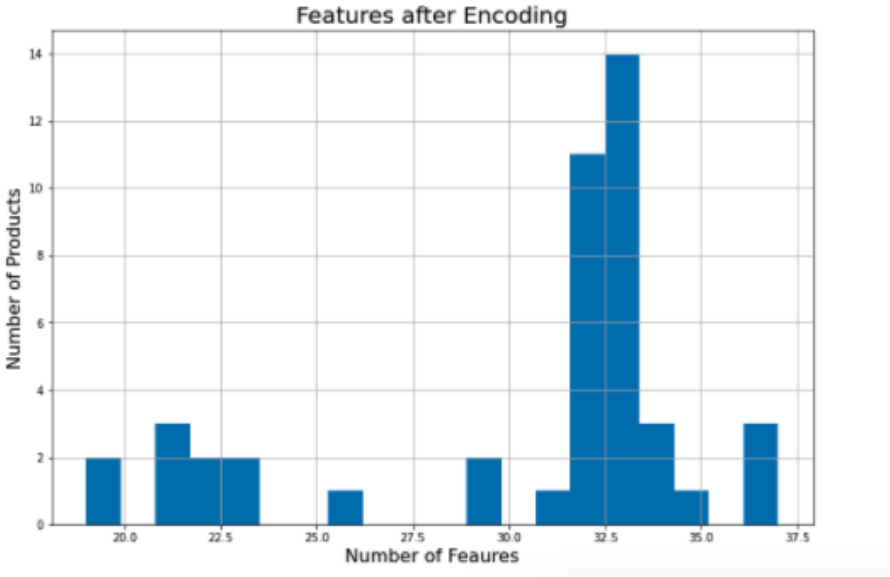

Figure 9. Number of Features after Encoding

ucts are having more than 30 features, indicating that DIF related features are largely enriching the original data-set on top of the four basic numerical features.

Data Normalization The purpose of data normalization is to increase the accuracy of the model, because normalization gives equal weights to each variable so that single variable in bigger numbers would not influence the model performance too much in one direction. In our experiments , z-score normalization (as known as standardization) has been applied to normalize the data. In practical, it means each variable would minus its mean value and divided by the standard deviation.

After the previous steps, the data preparation has completed and the data is ready for experimenting with the deep learning models.

3.2 Experiments

The experiments would be described from three perspectives: 1) the development environment where the experiments were performing, 2) the detailed implementation process and 3) the principles for evaluating the models.

3.2.1 Development environment

The development tools used in this study are listed as follows:

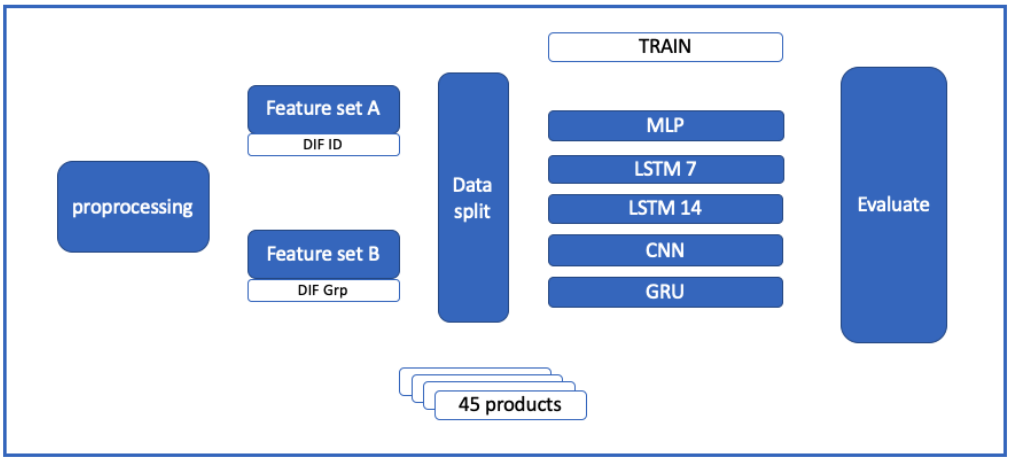

Figure 10. Implementation Pipeline

- Python programming environment 3.7.12
- Numpy, Pandas data science libraries for Python
- Scikit-learn machine learning library for Python
- Keras deep learning API for Python
- Matplotlib, plotly libraries for Data Visualization
- SAP Business Warehouse for data collection
- SAP Forecast and Replenishment for data collection

All machine learning experiments has been performed in Google Colaboratory.

3.2.2 Implementation

The implementation took place at the product level and each product would be trained by the five deep learning models. The implementation pipeline is illustrated in Figure 10. There are total 45 products in the data-set. Each product has been conducted with preprocessing activities and divided into two feature sets: feature set A includes DIF ID and feature set B include DIF Group, other features are exactly the same. Both feature sets are then split into 80% and 20% over the past 2 and half years time span for the training and the testing purposes, and feed to the deep learning models: MLP, CNN, LSTM and

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                 Output Shape              Param #
=================================================================
 dense (Dense)                (None, 100)               3400

 dense_1 (Dense)              (None, 1)                 101

=================================================================
Total params: 3,501
Trainable params: 3,501
Non-trainable params: 0
_________________________________________________________________
```

Figure 11. Model - Multilayer Perceptron

```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                 Output Shape              Param #
=================================================================
 conv1d (Conv1D)              (None, 28, 64)            192

 max_pooling1d (MaxPooling1D  (None, 14, 64)            0
 )

 flatten (Flatten)            (None, 896)               0

 dense_4 (Dense)              (None, 50)                44850

 dense_5 (Dense)              (None, 1)                 51

=================================================================
Total params: 45,093
Trainable params: 45,093
Non-trainable params: 0
_________________________________________________________________
```

Figure 12. Model - Convolutional Neural Network

GRU. Next we will go through the details of each model one by one.

Multilayer Perceptron - MLP MLP is used as baseline model due to its simple architecture. The model consists of two fully connected layers as shown in Figure 11, other parameters include activation ReLu and Dense 100 for the first layer, as well as learning rate 0.0003, optimizer Adam and epoches 60.

Convolutional Neural Network - CNN CNN is known for its powerful feature extraction capability by means of its Convolutional layer and Pooling layer. The detailed structure of the CNN model in this experiment can be found in figure 12.

```
Model: "sequential_12"
_________________________________________________________________
 Layer (type)                 Output Shape              Param #
=================================================================
 lstm_7 (LSTM)                (None, 50)                16800

 dropout_5 (Dropout)          (None, 50)                0

 dense_16 (Dense)             (None, 25)                1275

 dense_17 (Dense)             (None, 1)                 26

=================================================================
Total params: 18,101
Trainable params: 18,101
Non-trainable params: 0
_________________________________________________________________
```

Figure 13. Model - Long-short term Memory

LSTM LSTM is one advanced RNN which is good at dealing with time series problems. In our experiment, sliding window method is used to form the input and feed the model. Both window size 7 and 14 have been tried and dropout layer is added to generalize the calculation. The LSTM model structure is shown in Figure 13.

GRU Last but not the least, another advanced RNN model GRU is tested out. Two GRU layers and two dropout layers have been applied and followed by a dense layer at the end.(Figure 14)

As for now each product has been tried out with the models described above, now it is time to evaluate how these models have performed.

3.3 Evaluation

In this study the performance evaluation of the deep learning models has been conducted from two perspectives. Firstly, we have reviewed the Root Mean Square Error (RMSE) value of each experiment at the product and model level, aiming to measure the quality of the estimator by means of the deviation of the predicted and actual value. Second type of evaluation is taken care by a customized confusion matrix. The idea is to define a business oriented criteria to classify the predicted numeric value into positive or negative group and summarise the result in terms of a confusion matrix along with calculating the

```
Model: "sequential_5"
_________________________________________________________________
 Layer (type)                 Output Shape              Param #
=================================================================
 gru (GRU)                    (None, 29, 64)            12864

 dropout (Dropout)            (None, 29, 64)            0

 gru_1 (GRU)                  (None, 64)                24960

 dropout_1 (Dropout)          (None, 64)                0

 dense_8 (Dense)              (None, 1)                 65

=================================================================
Total params: 37,889
Trainable params: 37,889
Non-trainable params: 0
_________________________________________________________________
```

Figure 14. Model - GRU

relative scores, based on which we shall be able to see the performance of each model on the complete product list.

What is worth to mention is that during the implementation phase the time series type of data-set has been converted into supervised learning mode in the way that the label of the current day is shifted from the waste data of two days in future. The data-set has then been split into 80 and 20 percent for the training and testing purpose, which is to say 80 percent of the data has been used to train the model to predict what would be waste quantity in two days, and 20 percent of the data for calculating those performance indicators needed by the evaluations. More descriptions of the evaluation methods are explained in the following paragraphs.

3.3.1 RMSE

Root Mean Squared Error, or RMSE for short, is a standard way to measure the error of a model in predicting quantitative data. It is calculated as the square root of the mean of the squares of the predicted and actual values' deviations. The formula of RMSE is written in below:

$$RMSE = \sqrt{(\frac{1}{n}) \sum_{i=1}^{n} (y_i - x_i)^2} \tag{1}$$

In the formula above, xi denotes the actual value and yi represents the predicted value. In our particular case, RMSE is calculated within each experiment trial by adding up the deviation of every testing day's predicted and actual waste, and dividing by the number of the testing days. In this way we shall be able to get the average prediction variances of the particular model on each product.

3.3.2 Customized confusion matrix

From the store operation's perspective the workforce planning for the product waste inspection is arranged on the product level, which is to say the demand of the workload is calculated according to the number of the concerning single product unit. Therefore it would be more important to know what products might have potential waste to come, rather than how many pieces of box would be wasted. With this logic confusion matrix approach has been decided to use for prediction quality measurement, with the focus on model's ability to detect the waste than to estimate the exact waste quantity.

There are four indicators in a standard binary confusion matrix as shown in Figure 15 : True Positive (TP) - corrected predicted event values, True Negative (TN) - corrected predicted non-event values, False Positive (FP) - wrongly predicted event values,and False Negative (FN) - wrongly predicted non-event values.

The output of our deep learning models is waste value in numeric format as the problem itself has been handled as a regression one. To convert a regression problem into a classification one, proper rules should be defined to categorize the numeric predicted value into either positive or negative class. The confusion matrix used in the model evaluation of this study has been defined as follows. In the operative circumstances the minimum waste quantity is 1 in case at least one box is expired or damaged, so when the actual value is equal or greater to 1, it is considered as positive. On the other hand, when classifying the predicted value, it is at first being compared with certain threshold, if predicted value is greater than the threshold it is classed as positive meaning that the model predicts the food waste will happen in two days. Such prediction is considered as correct if the actual waste happens in at least once within the future three days. The complete definition of the confusion matrix has been listed in Table 2.

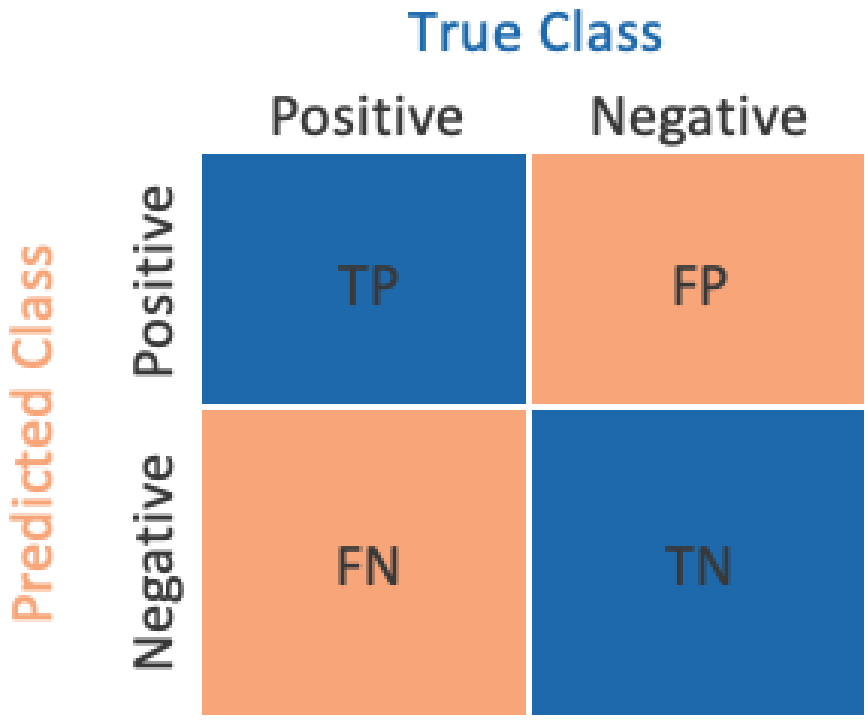

Figure 15. Confusion Matrix

Table 2. Confusion Matrix Definition

Confusion Matrix Definition				
Category	Pred Value (Day X)	Actual Value (Day X-1)	Actual Value (Day X)	Actual Value (Day X+1)
TP	>= threshold	OR >=1	OR >=1	OR >=1
TN	< threshold	AND <1	AND <1	AND <1
FP	>= threshold	AND <1	AND <1	AND <1
FN	< threshold	OR >=1	OR >=1	OR >=1

As illustrated in Table 2, on Day X if the predicted value is greater than or equal to the threshold, and the actual waste has happened either on Day X ,Day X+1, or Day X+2, the prediction is classified as True Positive (TP). Under the same circumstance if none waste happened on Day X, Day X+1, or Day X+1, the prediction is False Positive (FP). In addition, on Day X if the predicted value is less than the threshold, and the actual waste has happened either on Day X ,Day X+1, or Day X+2, the prediction is classified as False Negative (FN). Otherwise False Positive(FP) would be marked if the predicted value is less than the threshold, and the actual waste has not happened at all during these three days. The intuitive behind the confusion matrix is that if the food waste would come in the future 1 or 2 or 3 days and we are able to predict it correctly today, the prediction can be considered as valuable because business would get at least 1 day to plan for the potential waste in advance.

Apart from calculating the number of true or false classification, the following scores associated with confusion matrix have also been calculated for each product and model combination. They are accuracy, recall, precision and F1 score.

Accuracy Accuracy represents the number of correctly classified data instances over the total number of data instances. It can be calculated in the formula below.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

Accuracy might not be the best measure when the data-set is imbalanced. As stated in 3.1.2, our data-set is imbalanced because most products have much more negative values than positive. In such scenario even the model failed to predict the positive value, the accuracy score can still be high.

Precision Precision also called positive predictive value, is defined as the ratio of correct positive predictions to the total predicted positives. Its calculation is also expressed in the following formula.

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

Precision is an appropriate performance indicator when minimizing false positives is the focus.

Recall Recall, as known as sensitivity or true positive rate, is defined as the ratio of correct positive predictions to the total positives examples.

$$Recall = \frac{TP}{TP+FN} \tag{4}$$

Recall is more appropriate when minimizing false negatives is the focus.

F1 Score F1-score is a metric which takes into account both precision and recall and is defined as follows:

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} = \frac{2 * TP}{2 * TP + FP + FN} \tag{5}$$

F1-score has combined both precision and recall, thus is a better measure than accuracy especially for imbalanced data-set.

Due to different consumption and replenishment pattern, it is not likely that one model would behave the best on all the products. Some products might work better with model A, and some performs better with model B. With the evaluation approaches defined so far, we shall be able to evaluate the quality of each experiment trial as well as overall behavior of each model.

4 RESULTS

In this section, we will go through the results of the experiments according to the evaluation methods defined in the previous chapter.

The outcome of totally 1350 experiments have been recorded which concerns:

- 45 Products
- 2 Feature Sets
- 5 Deep Learning Models
- 3 Thresholds

The results have been documented in a two dimensional table as is illustrated in Figure 16. As shown, each row is associated with one particular trial on the product and model

	article	model	feature_set	rate	val_loss	train_rmse	number of features	number of records	number of test	TP	TN	FP	FN
0	20006270	MLP	DIF ID	0.8	0.758799	0.802666	37	1017	203	3	139	48	11
1	20006270	MLP	DIF ID	0.9	0.758799	0.802666	37	1017	203	3	147	40	11
2	20006270	MLP	DIF ID	1.0	0.758799	0.802666	37	1017	203	2	153	34	12
3	20007204	MLP	DIF ID	0.8	1.894309	0.853268	32	1016	202	1	170	21	8
4	20007204	MLP	DIF ID	0.9	1.894309	0.853268	32	1016	202	1	174	17	8
...	...	...	...	...	...	...	...	...	...	...	...	...	...
1345	21507740	GRU	DIF Group	0.9	0.392555	0.904445	20	1016	202	1	167	15	17
1346	21507740	GRU	DIF Group	1.0	0.392555	0.904445	20	1016	202	1	172	10	17
1347	21598762	GRU	DIF Group	0.8	0.434490	0.991099	19	955	190	1	152	12	23
1348	21598762	GRU	DIF Group	0.9	0.434490	0.991099	19	955	190	0	164	0	24
1349	21598762	GRU	DIF Group	1.0	0.434490	0.991099	19	955	190	0	164	0	24

1350 rows × 13 columns

Figure 16. Experiment Result

bases, which can be identified by the first two columns: the 'article' column contains the product code used in the case company's ERP system, and the 'model' column indicates the corresponding ML model name. 'feature_set' indicates either DIF ID or DIF Group is included to the feature selection and 'rate' column tells the threshold used in the confusion matrix classification. Column 'val_loss' and 'train_rmse' stored the RSME value of the training and testing data, and the confusion matrix related indicator: 'TP','TN','FP','FN' are also being counted and saved. In addition the number of the features and the size

of the data points are also included to the result table. More implications behind the numbers are going to be explored as follows. We will use similar approach to go through the outcome of the evaluations, in terms of checking the value distribution via box plot together with the key statistic summary, aiming to find out the overall performance of the deep learning models on the target data-set, as well as the similarity and difference among the models.

4.1 RMSE

Root Mean Squared Error, as known as RMSE provides a straight forward measure of the difference between the predicted and actual value. The less the RMSE value is, the better the estimator works. RMSE becomes zero when the predicted value is exactly the same as the actual value.

In Figure 17, boxplots are used to provide an overview of the RMSE value distribution achieved by the five models on the 45 products, and color code is used to mark each Feature Set. As seen, the shape of the boxplot is quite similar in all models, where the majority of the products have RMSE distributed between 0 and 1, and the outlier's are only found beyond the max value of the boxes. Feature set DIF Group's RSME value is in general smaller than Feature set DIF ID for most of the products, because the Median line and the IQR box of the former are mostly closer to zero than the latter. When do the comparison across the models, GRU and MLP have got smaller RMSE median and IQR value, which indicates the predictions made by these two models are more accurate for most of the products than the rest models. On the other hand, LSTM models seem to have wider IQR and couple outlier's in extreme big value, which tells LSTM model has not worked well in few specific products.

Figure 18 has provided the key statistic figures of RMSE value grouped by model name and Feature Set as supplementary information. According to Figure 18, we can see the median, average, max and min RMSE value of all the 45 products per each model, where model GRU has been observed to outperform the others in terms of the lowest median value (0.44) and mean value (0.52). MLP as the baseline model, surprisingly ranks at the second place even with its simple architecture. The predictions made by CNN and

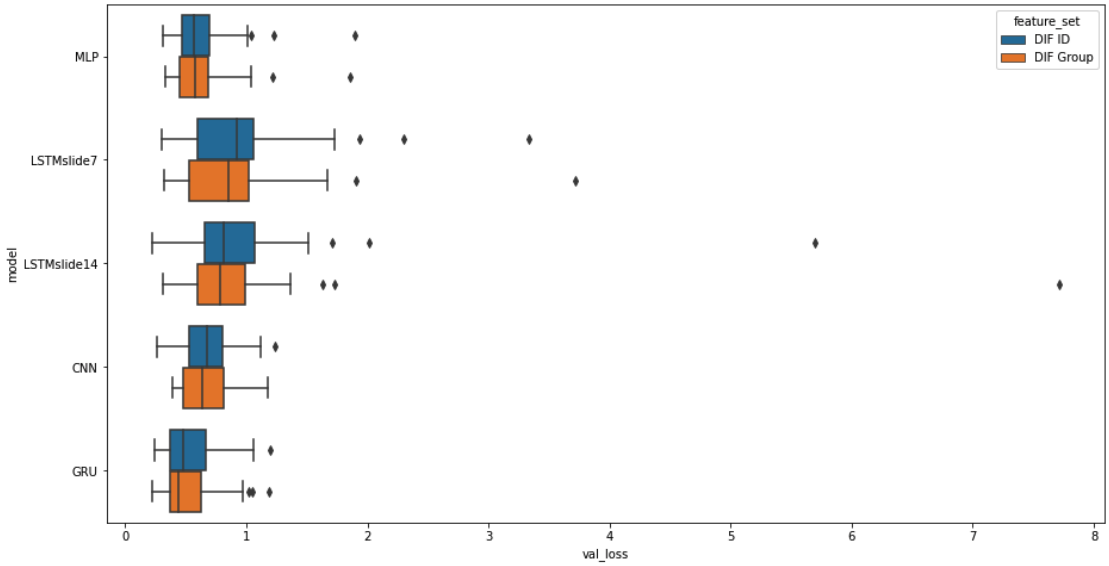

Figure 17. RMSE Distribution

model	feature_set	count	mean	std	min	25%	50%	75%	max
CNN	DIF Group	45.0	0.664080	0.215412	0.386015	0.483176	0.638112	0.811204	1.178259
	DIF ID	45.0	0.681391	0.215918	0.261359	0.524184	0.678672	0.807957	1.237112
GRU	DIF Group	45.0	0.526165	0.231393	0.222497	0.367702	0.444490	0.627662	1.188990
	DIF ID	45.0	0.533495	0.219382	0.238802	0.368474	0.483957	0.664917	1.201183
LSTMslide14	DIF Group	45.0	0.967318	1.074034	0.307823	0.601729	0.788304	0.993121	7.710721
	DIF ID	45.0	0.996382	0.795584	0.220654	0.658402	0.816514	1.071493	5.695226
LSTMslide7	DIF Group	45.0	0.888964	0.547009	0.317277	0.532015	0.852279	1.024047	3.719803
	DIF ID	45.0	0.967423	0.554134	0.304728	0.599691	0.924776	1.064477	3.336760
MLP	DIF Group	45.0	0.626301	0.272640	0.328410	0.450156	0.572915	0.690519	1.860688
	DIF ID	45.0	0.638802	0.275368	0.310417	0.469262	0.563002	0.696065	1.894309

Figure 18. RMSE Statistics

LSTM models on the other hand have turned out to deviate more with the actual values comparing to their peers. The max RMSE achieved by LSTM model is found to be much higher (7.71) than the rest models (between 1 and 2).

Figure 19 provided a third angle, showing the number of models who has achieved the lowest RMSE value for each product. From this perspective GRU has again been performing the best in 28 products out of 45, which is about 62% of the total products in scope. Second is MLP which is working best for 7 products, and followed by LSTM models and CNN, which respectively suits best for 5,4 and 1 product.

GRU	28
MLP	7
LSTMslide7	5
LSTMslide14	4
CNN	1

Figure 19. RMSE - Best Model

In short, the analysis of RMSE can be concluded that all five deep learning models have shown a similar pattern of RMSE value distribution where the data range is between 0 and 1, Median and IQR skewed to zero, and outlier's only exist beyond max value. Feature set DIF Group has got smaller RMSE mean value than DIF ID for all the models, which might suggest that generalized DIF can be easier for the models to learn the waste pattern. Among the models studied, GRU model has performed the best from the perspective of RMSE, based on the fact that GRU achieves smallest RMSE value in most of the products in concern.

4.2 Customized Confusion Matrix

As the second view of the evaluation process, we will assess the models from the perspective of a customized confusion matrix. Feature Set DIF Group has performed better than DIF ID according to the findings of RMSE analysis, thus Feature set DIF Group has been used in the confusion matrix analysis. As explained earlier, each experiment trial has been calculated with its own TP, TN, FP and FN as per three thresholds - 0.8, 0.9, 1, based on which the Confusion matrix relevant scores Accuracy, Precision, Recall and F1 have been further calculated, and will be reviewed in this part.

4.2.1 Accuracy

Accuracy is calculated by means of dividing the total correctly predicted value (TP+TN) by the total predicted value (TP+TN+FP+FN), therefore Accuracy score takes positive and negative value into consideration with equal importance. The range of Accuracy is between 0 and 1 and the best score can be up to 1 when the predictions are 100 % correct.

The distribution of each models' Accuracy score has been illustrated in Figure 20 where the color indicates the specific rate being used. Many similarities are found to be shared

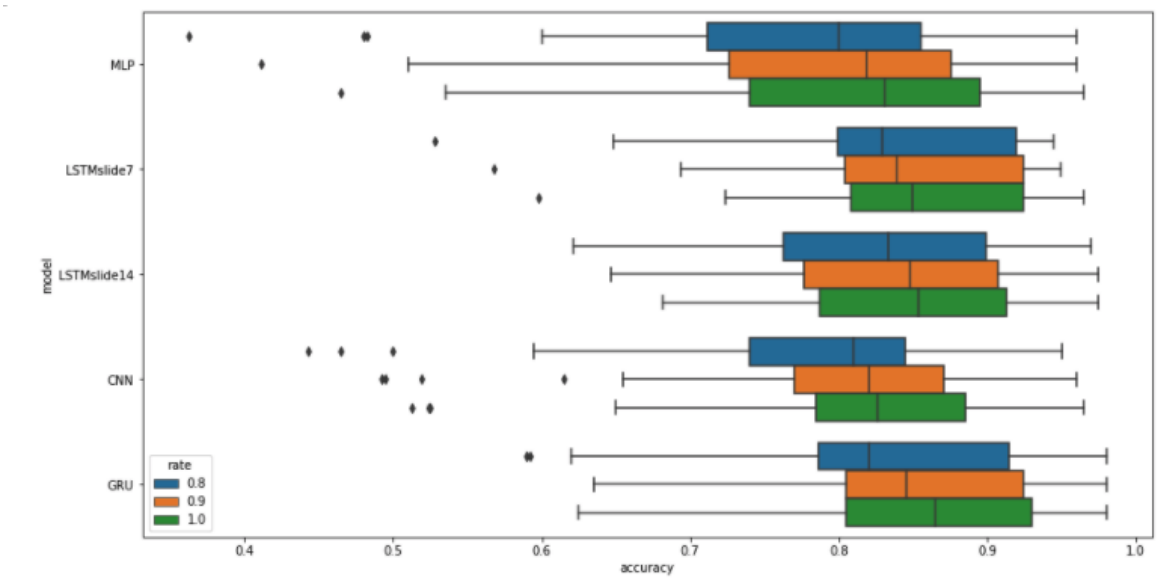

Figure 20. Accuracy Distribution

among all the models, for instance the greater value the rate is, the greater median and IQR locates. In addition, the outlier's only exist outside of the left whisker of the box plots for some of the models, while there is no outlier's at all beyond the right whiskers . The median value of the Accuracy score is around 0.83 for all the models.

Aligned with the previous finding in RMSE analysis, when compare the performances among the models, GRU again has outperformed the rest with regard to having higher Accuracy score for most products. LSTM models are next to GRU which have relatively high median and IQR value but also have few outlier's with low value. Regarding MLP and CNN models, most of the products have got lower Accuracy scores with them.

The statistics of the Accuracy score per each model and rate has been further listed in Figure 21. As seen in Figure 21, Accuracy is generally higher in case of rate 1 than rate 0.8 and 0.9. The median value is between 0.81 and 0.87 and mean value between 0.78 and 0.85 across the various combinations of model and rate. Among the concerning five models GRU has achieved highest median and mean Accuracy scores, for example its median accuracy value has reached 0.87 and mean value 0.85. The median and mean value of LSTM models are slightly lower than GRU, with mean and median value around 0.85. MLP and CNN has got bigger difference and got mean value near 0.81 and median value 0.83.

		count	mean	std	min	25%	50%	75%	max
model	rate								
CNN	0.8	45.0	0.784414	0.115530	0.442786	0.740000	0.810000	0.845000	0.950249
	0.9	45.0	0.799760	0.110341	0.492537	0.770000	0.820000	0.870647	0.960199
	1.0	45.0	0.810209	0.106138	0.512438	0.785000	0.825871	0.885000	0.965174
GRU	0.8	45.0	0.831912	0.096475	0.590000	0.786070	0.820000	0.915000	0.980100
	0.9	45.0	0.846404	0.088540	0.635000	0.805000	0.845771	0.924623	0.980100
	1.0	45.0	0.854377	0.086213	0.625000	0.805000	0.865000	0.930000	0.980100
LSTMslide14	0.8	45.0	0.832121	0.084842	0.621212	0.762626	0.833333	0.898990	0.969697
	0.9	45.0	0.842302	0.080710	0.646465	0.776650	0.847716	0.907692	0.974747
	1.0	45.0	0.849152	0.075503	0.681818	0.786802	0.853535	0.912821	0.974747
LSTMslide7	0.8	45.0	0.836566	0.088124	0.527638	0.798995	0.829146	0.919598	0.944724
	0.9	45.0	0.843832	0.081481	0.567839	0.804020	0.839196	0.924623	0.949749
	1.0	45.0	0.851146	0.076415	0.597990	0.808081	0.849246	0.924623	0.964824
MLP	0.8	45.0	0.775263	0.128691	0.362162	0.711443	0.800000	0.855000	0.960199
	0.9	45.0	0.793561	0.122890	0.410811	0.726368	0.819095	0.875622	0.960199
	1.0	45.0	0.808449	0.116073	0.464865	0.740000	0.830846	0.895000	0.965174

Figure 21. Accuracy Statistics

In summary, the analysis of the Accuracy score has revealed that the Accuracy score grows while the threshold rate increases, and most products have achieved Accuracy score above 0.8. GRU and LSTM models behave better than CNN and MLP in terms of high median and mean Accuracy value.

What is worth mentioning is that due to the imbalance nature of the data-set where negative value is much more than the positive value, the accuracy score might not provide a good insight on how well the model is in fact able to predict the positive value. The upcoming analysis of scores Precision, Recall and F1 shall shed more light on this regard.

4.2.2 Precision, Recall, F1

Precision, Recall and F1 scores are important measurements when the positive class is the focus than the negative. Precision refers to the percentage of the correctly predicted positive class (TP) over the number of total positively predicted value (TP+FP), and Recall is defined as the ratio of correct positive predictions(TP) to the total positives examples (TP+FN). Precision is a good indicator when minimizing the false positive is the focus while Recall is wise to check when avoiding the false negative is more crucial. Regarding the business case of this study Recall is more relevant as the goal is to predict as much as possible upcoming food waste - the positive class. F1 score is the harmonic mean of Precision and Recall as it takes both scores into account, therefore F1 score is a more reliable indicator than Accuracy when dealing with the unbalanced data-set. The value of all three scores range from 0 to 1 , and the best score can be 1 in case of no false positive or false negative values predicted. On the contrary, if scores are calculated to be or near zero it means the number of correctly predicted positive value is rather limited.

We will first check the overall value distribution of the three scores in the box plot shown in Figure 22.

The three charts in Figure 22 are respectively for Precision, Recall and F1 score, and each chart contains five box plots for the five studied models. Again color code is used here to tell the classification threshold, also the same X axis is shared by three charts to facilitate the score comparison. In general, the most products' three scores are between 0 and 0.4.

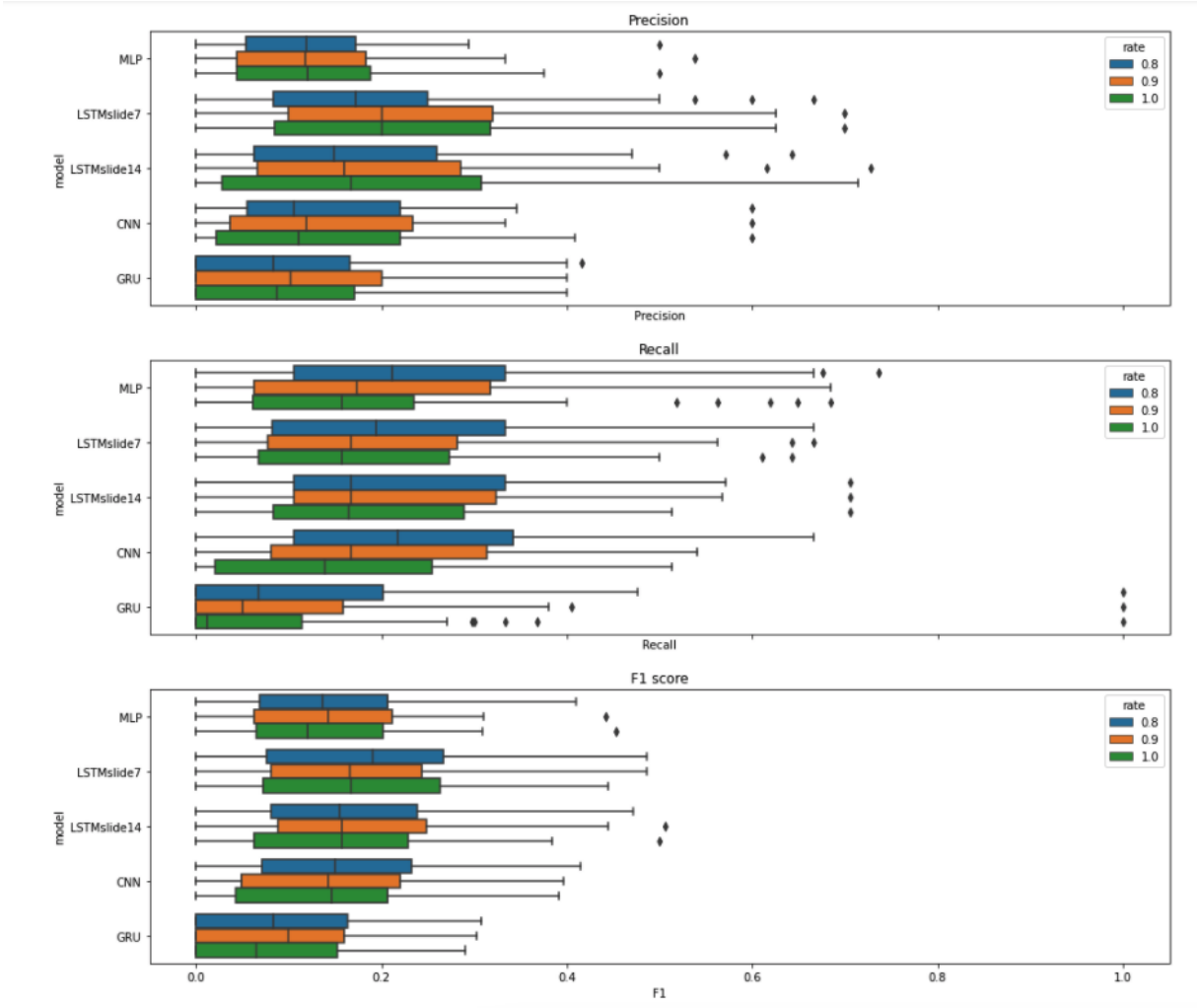

Figure 22. Precision, Recall and F1 Distribution

		Precision								Recall								F1							
		count	mean	std	min	25%	50%	75%	max	count	mean	std	min	25%	50%	75%	max	count	mean	std	min	25%	50%	75%	max
model	rate																								
CNN	0.8	44.0	0.145775	0.124616	0.0	0.055481	0.105572	0.220395	0.600000	44.0	0.242835	0.180948	0.0	0.106061	0.218254	0.342636	0.666667	43.0	0.156067	0.103529	0.0	0.070802	0.150000	0.231933	0.415094
	0.9	44.0	0.145134	0.131459	0.0	0.036944	0.118789	0.234355	0.600000	44.0	0.198552	0.163724	0.0	0.080357	0.166667	0.314103	0.540541	43.0	0.141728	0.103587	0.0	0.048897	0.142857	0.219807	0.396040
	1.0	44.0	0.139324	0.133883	0.0	0.022059	0.111111	0.220190	0.600000	44.0	0.169733	0.160318	0.0	0.020833	0.138996	0.254808	0.513514	43.0	0.130865	0.103696	0.0	0.043056	0.146341	0.207143	0.391753
GRU	0.8	40.0	0.111363	0.118843	0.0	0.000000	0.083916	0.165799	0.416667	44.0	0.139152	0.189350	0.0	0.000000	0.066964	0.201897	1.000000	39.0	0.101256	0.089688	0.0	0.000000	0.083333	0.163839	0.307692
	0.9	36.0	0.128872	0.125632	0.0	0.000000	0.101471	0.200000	0.400000	44.0	0.114940	0.184557	0.0	0.000000	0.050641	0.158854	1.000000	35.0	0.100889	0.086940	0.0	0.000000	0.100000	0.159825	0.302326
	1.0	36.0	0.105197	0.112495	0.0	0.000000	0.087121	0.170833	0.400000	44.0	0.092675	0.175749	0.0	0.000000	0.012500	0.113636	1.000000	35.0	0.082167	0.084638	0.0	0.000000	0.064516	0.152074	0.289855
LSTMslide14	0.8	45.0	0.178902	0.157013	0.0	0.062500	0.148148	0.259259	0.642857	44.0	0.215527	0.177258	0.0	0.106061	0.166667	0.333333	0.705882	44.0	0.171520	0.130583	0.0	0.080842	0.155354	0.239191	0.471910
	0.9	45.0	0.197958	0.175432	0.0	0.066667	0.160000	0.285714	0.727273	44.0	0.207587	0.171987	0.0	0.106061	0.166667	0.323333	0.705882	44.0	0.173680	0.130425	0.0	0.088095	0.157379	0.248891	0.506024
	1.0	45.0	0.194976	0.177250	0.0	0.027778	0.166667	0.307692	0.714286	44.0	0.184447	0.158705	0.0	0.083333	0.164414	0.289286	0.705882	44.0	0.160400	0.121771	0.0	0.062500	0.156923	0.228635	0.500000
LSTMslide7	0.8	45.0	0.203967	0.175300	0.0	0.083333	0.172414	0.250000	0.666667	44.0	0.219039	0.174833	0.0	0.081731	0.193750	0.333333	0.666667	44.0	0.178250	0.124067	0.0	0.075625	0.190909	0.267073	0.486486
	0.9	45.0	0.210302	0.179867	0.0	0.100000	0.200000	0.320000	0.700000	44.0	0.193896	0.167813	0.0	0.076923	0.166667	0.282366	0.666667	44.0	0.169624	0.116582	0.0	0.080842	0.165525	0.243357	0.486486
	1.0	44.0	0.214654	0.177285	0.0	0.084848	0.200000	0.317708	0.700000	44.0	0.177996	0.162544	0.0	0.066952	0.157072	0.273015	0.642857	43.0	0.166544	0.116053	0.0	0.072917	0.166667	0.262726	0.444444
MLP	0.8	44.0	0.126725	0.100118	0.0	0.053571	0.119639	0.171781	0.500000	44.0	0.246189	0.196691	0.0	0.106061	0.211111	0.333333	0.736842	43.0	0.147121	0.097426	0.0	0.068287	0.136364	0.206061	0.409836
	0.9	44.0	0.127837	0.110913	0.0	0.044091	0.117460	0.182692	0.538462	44.0	0.215885	0.189689	0.0	0.062831	0.173077	0.317708	0.684211	43.0	0.138872	0.102822	0.0	0.062996	0.142857	0.211982	0.442478
	1.0	43.0	0.131581	0.116847	0.0	0.044466	0.120000	0.188483	0.500000	44.0	0.191414	0.183524	0.0	0.061147	0.156923	0.235577	0.684211	42.0	0.136134	0.105234	0.0	0.065336	0.120072	0.201143	0.452830

Figure 23. Precision, Recall and F1 Key Info

It can be also observed that smaller threshold has lead to smaller Precision score, but bigger Recall score. F1 is determined by both Precision and Recall, and the best F1 score is achieved by LSTM model with window size 7, in case of threshold 0.8. Now let's zoom into the performance of each model. In the earlier result review, GRU model has got best performance in terms of RSME and Accuracy score, however it behaves the opposite way in the Presicion, Recall and F1 score, with lowest median and IQR among all models. The rest four models on the other hand have had similar behavior in Recall score distribution, while LSTM models have got higher median Precision scores than others.

The key statistic figures of the three scores grouped by model and rate are listed in Figure 23, which includes the count, mean, standard deviation, minimum, quartile and max value of the specific score per each model. There are 45 products being evaluated in the confusion matrix. However the counts of all scores are under 45 because Precision score is not available in case TP+FP is zero for certain products and similarly Recall score cannot be calculated if TP+FN is zero. F1 score depends on both Precision and Recall therefore it has value in case the other two scores are relevant.

When browsing further over the mean and quartile values in the table, it can be observed that the scores vary from model to model but the overall absolute mean and median value are pretty low. The mean Precision and Recall scores of the five models are around 0.2, while F1 is slightly lower and between 0.08 and 0.17. In addition, the minimum value of the three scores is 0 for all models, which indicates that deep learning models in study do not work well on at least some products to predict correctly any waste. If we compare the

scores cross the models, GRU has got lowest mean and median value in all three scores, but it also achieved the highest maximum Recall score 1 which none of the others can do. LSTM models on the other hand have got better scores in the evaluation of Precision, Recall and F1 scores.

The analysis of Precision, Recall and F1 has reached a complete different conclusion than the previous analysis of RMSE and Accuracy. The RMSE and Accuracy score in general are in decent level over for almost all the models, where GRU model has ranked the best among the studies models. However according to the Precision, Recall and F1 scores which are more suitable for describing the imbalanced data-set, the quality of predictions made by the deep learning models are not very satisfactory as the majority of the products got these three scores close or near to 0.2.

5 CONCLUSIONS

5.1 SUMMARY

In this thesis we have experimented four deep learning models MLP, CNN, LSTM and GRU on a time series data-set, aiming to find out how well the deep learning models are able to predict the potential food waste in the near future. The data-set consists of historical data of the Ready-to-Eat products from a grocery store in a Finland, across the time period from Jan 2019 to Sep 2021. After data cleaning, 45 products with enough waste record are retained for the training and testing. The performance of the experiments have been evaluated by means of RMSE value and a customized confusion matrix. Most products have got RMSE value between 0 and 1 for all models, where model GRU and MLP have achieved the smaller mean and median RMSE compared the others. Regarding confusion matrix related score measurements, the majority of the products have got Accuracy score over 0.8 and the Median value of Precision, Recall and F1 scores are near 0.2. In addition LSTM models have been observed slightly better than the peers concerning the confusion matrix related performances. GRU on the other hand are lagging behind on this regard.

5.2 Future work

Due to the limited time and data, the study has been limited to one store and one product group. It might be worth to extend the research to more stores and product groups, so as to have more comprehensive view on the capability of the deep learning models in this business context. Moreover, traditional machine learning regression algorithms e.g. XGBooster, Random Forest are also good to try on the same data-set to see the pros and cons compare to deep learning models.

REFERENCES

Abirami, S. & Chitra, P. 2020, Chapter Fourteen - Energy-efficient edge based real-time healthcare support system, In: Pethuru Raj & Preetha Evangeline, eds., *The Digital Twin Paradigm for Smarter Systems and Environments: The Industry Use Cases*, *Advances in Computers*, vol. 117, Elsevier, pp. 339–368. Available: https://www.sciencedirect.com/science/article/pii/S0065245819300506.

2021, anualreport. Available: https://www.kesko.fi/globalassets/03-sijoittaja/raporttikeskus/2021/q1/kesko_annual_report_2020_sustainability.pdf.

Bahi, Meriem & Batouche, Mohamed. 2018, Deep Learning for Ligand-Based Virtual Screening in Drug Discovery.

Blaji, Sai. 2020, DBinary Image classifier CNN using TensorFlow. Available: https://medium.com/techiepedia/binary-image-classifier-cnn-using-tensorflow-a3f5d6746697.

Dinesh C.S. Bisht, Mangey Ram. 2021.

FAO. 2011, Global Food Losses and Food Waste. Extent, Causes and Prevention.

Felicitas Schneider, Mattias Eriksson. 2020, FOOD WASTE (AND LOSS) AT THE RETAIL LEVEL.

2021, GROCERY TRADE. Available: https://www.kesko.fi/en/company/divisions/grocery-trade/.

IBM. 2021, What is Machine learning. Available: https://www.ibm.com/cloud/learn/machine-learning.

Lars Kegel, Martin Hahmann & Lehner, Wolfgang. 2018.

Lek, S. & Park, Y.S. 2008, Multilayer Perceptron, In: Sven Erik Jørgensen & Brian D. Fath, eds., *Encyclopedia of Ecology*, Oxford: Academic Press, pp. 2455–2462. Available: https://www.sciencedirect.com/science/article/pii/B9780080454054001622.

Phi, Michael. 2018, *Illustrated Guide to LSTM's and GRU's: A step by step explanation*. Available: https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21.

Rivas, Dr. Pablo. 2020, *Deep Learning for Beginners*, Packt Publishing.

S. Hochreiter, J. Schmidhuber. 1997.

Samuel, A. L. 2000, Some studies in machine learning using the game of checkers, *IBM Journal of Research and Development*, vol. 44, no. 1.2, pp. 206–226.

Selvin, Sreelekshmy; Vinayakumar, R; Gopalakrishnan, E. A; Menon, Vijay Krishna & Soman, K. P. 2017, Stock price prediction using LSTM, RNN and CNN-sliding window model, In: *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 1643–1647.

Stenmarck, Jensen C. Quested T. Moates G., Å. 2016, Estimates of European food waste levels.

UNEP. 2021, UNEP Food Waste Index Report 2021.