CLASSIFICATION OF FINANCIAL NEWS USING SENTIMENT ANALYSIS TECHNIQUES

SENTIMENT ANALYSIS ON FINANCIAL NEWS

DAN CALIN ANDREI MINCA

THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
BACHELOR OF SCIENCE IN COGNITIVE SCIENCE & ARTIFICIAL INTELLIGENCE

DEPARTMENT OF
COGNITIVE SCIENCE & ARTIFICIAL INTELLIGENCE
SCHOOL OF HUMANITIES AND DIGITAL SCIENCES
TILBURG UNIVERSITY

STUDENT NUMBER

2016744

COMMITTEE

dr. Mirella de Sisto
dr. Paula Roncaglia

LOCATION

Tilburg University
School of Humanities and Digital Sciences
Department of Cognitive Science &
Artificial Intelligence
Tilburg, The Netherlands

DATE

June 28, 2022

ACKNOWLEDGMENTS

Firstly I would like to thank my thesis supervisor dr. Mirella de Sisto for her guidance and constant support over the course of the past months. I would also like to thank dr. Sharon Ong for her valuable advice. Lastly, I would like to thank all the thesis supervisors that took part at the poster presentation and offered me very relevant feedback that allowed me to set the path for this project.

CLASSIFICATION OF FINANCIAL NEWS USING SENTIMENT ANALYSIS TECHNIQUES

SENTIMENT ANALYSIS ON FINANCIAL NEWS

DAN CALIN ANDREI MINCA

Abstract

Sentiment analysis is the statistical analysis of simple sentiment cues. Essentially, it involves making statistical analyses on polarized statements (i.e., statements with a positive, negative and neutral sentiment), which are usually collected in the form of social media posts, reviews, and news articles. In recent times, significant developments in this field have resulted in a wealth of research towards sentiment analysis techniques. This has led to a divergence in approaches when addressing sentiment analysis tasks – with both machine learning and deep learning models being used to perform such tasks. This thesis chooses to focus on this divergence of approaches in solving a sentiment analysis task – with the primary aim being to perform a comparative analysis on the performance of both machine learning and deep learning models for a sentiment analysis task. Owing to the complexity of deep learning models in general, we choose only one model – that of a recurrent neural network (RNN), in the form of a Gated Recurrent Unit (GRU). The performance of this model is compared and contrasted with that of several popular machine learning models used for such tasks, namely; Naïve Bayes(NB), Support Vector Machine (SVM) and Logistic Regression (LR). There are two important considerations that must be taken into account while performing such a comparative analysis – the dataset selected and the nature of the model chosen. The results of the study indicate that the Logistic Regression model is the best performing model across all combinations of the chosen dataset.

1 INTRODUCTION

One of the most relevant factors that influence the stock market is news articles. In fact, there is a strong, but complicated relation between the information from news and the market movement i.e, the volatility of various stock prices (Kirange & Deshmukh, 2016). Investors are gaining

real-time information on stock market predictions in the form of financial news and try to decide on their investment strategies on the basis of these predictions. This can mean at the same time maximizing profits, or reducing losses, depending on how the market reacts to the news. Therefore, being able to accurately classify this news in real time, becomes of great importance for investors and their assets.

The financial market is a very complex system with a lot of inter-dependencies. Because of this complexity, in combination with the enormous amount of data available in the online environment, most classical economic models fail to explain the dynamics in the market. Thus, in recent years there has been a substantial growth in interest and research for more data-driven methods that incorporate computational techniques (Wan et al., 2021).

The statistical analysis of relatively simple sentiment cues can provide a surprisingly meaningful sense of how the latest news impacts important entities (Godbole, Srinivasaiah, & Skiena, 2007) . Attempts to classify financial news in real time usually involve the technique of sentiment analysis - which is the process of detecting the polarity of a given text. This technique is also known as opinion mining and is concerned with identifying and extracting affective states and meaning from people's opinion – some example of how this can be represented include social media posts, news articles, and movie reviews. Sentiment analysis is multidisciplinary in nature, as it can incorporate approaches from various disciplines, such as NLP (Natural Language Processing), machine learning or computational linguistics. In the last decade, there was a significant increase in research towards the field of sentiment analysis. This was mostly due to its ability to accurately analyse and classify huge amount of data in a matter of seconds, with good accuracy (Baid, Gupta, & Chaplot, 2017).

Gaining information from news was always a very popular method, but as of late, with the recent growth of online platforms, the volume of the news has also increased rapidly. The rise of social media networks has significantly impacted the growth of online news sources. The need for convenient access to news sources has always been in high demand, and this is no different in the digital age, with a large number of investors turning to social media for news sources. Additionally, investors are increasingly falling victim to fraudulent financial news articles available online. These concerns are linked, because although crowd-sourced outlets can lower the cost of information acquisition and speed its dissemination, they also provide a venue for interested parties to spread fake information in an attempt to manipulate the markets (Kogan, Moskowitz, & Niessner,

2019) . Thus, it is becoming increasingly difficult for investors to judge the relevance of a financial news articles.

The stock market is a very volatile place. Price fluctuations are extremely common and even constant in some financial investments (for example, in certain commodities, and also cryptocurrency) and there are a multitude of factors that can influence the direction of the price movement in the stock market. People's sentiment and perception towards a specific company can change at any time, if presented with relevant information. Therefore, investors in the stock market heavily rely on the insights acquired from financial news articles.

In this thesis, several different classification algorithms will be trained, and subsequently compared with the aim of determining which of the presented models is best suited for the task of classifying financial news articles using the technique of sentiment analysis. The objective will be to compare the performance of these algorithms based on various evaluation metrics such as precision, recall, accuracy and f1 score, metrics that will be computed based on the true positive, true negative, false positive and false negative values. In performing this comparison, our aim is to identify the characteristics that enable a model to better perform the task of sentiment analysis, which would be beneficial for the development and design of future models.

Although this project is going to focus strictly on automating the process of identifying and extracting affective states from financial news, the techniques and results discussed here can be applied towards other services that involve the same type of problems, such as detection of inflammatory / hate speech in online platforms; marketing research; or in any type or recommendation system.

In addition, manually classifying the enormous amount of financial news that is being published every day is a very difficult and exhausting task (Yadav, Jha, Sharan, & Vaish, 2020). Moreover, not only is it difficult, but the evaluation itself may not be completely objective due to factors such as personal beliefs, fatigue, bias or different emotions. These limitations can be overcome to a reasonable extent by automating this process. Furthermore, identifying important features in successful classifiers will benefit the future development of models used for this process. For the purposes of this thesis, the following classifiers will be utilized: Naïve Bayes(NB), Logistic Regression(LR), Support Vector Machine (SVM) and Recurring Neural Network(RNN) in the form of a Gated Recurrent Unit (GRU).

The results obtained from evaluating the selected models will be compared against benchmark results obtained in earlier studies, such as (Malo, Sinha, Takala, Korhonen, & Wallenius, 2013). The dataset used in this

project was already annotated by 16 human judges with adequate backgrounds in the financial markets (Malo et al., 2013) and therefore it will be used as the ground truth labelled data that will serve for metric analysis.

Central Research Question:

•From a selection of machine learning and deep learning models, which one yields the best results when classifying financial news articles based on their polarity?

Subsequently, the main research question will be addressed through the following sub-questions:

•How well do machine learning and deep learning models classify text data to predict on the 'sentiment' or affective state behind the message from news article?

• Which model provides the best results while performing sentiment analysis on financial news articles?

2 RELATED WORK

At a very fundamental level, the field of sentiment analysis focuses on determining the polarity of a given text (positive or negative). Previous research on this topic showed that there are multiple approaches available to work on sentiment analysis, such as: lexical affinity, keyword spotting, statistical methods and concept-level techniques (Cambria, Schuller, Liu, Wang, & Havasi, 2013). Moreover, Märkle-Huß, Feuerriegel, and Prendinger (2017) explained in more detail techniques such as Bag of Words, that takes into account the frequency of each word, and their n-gram combinations, while also arguing that previous study did not account for the relationship between the document structure and the semantic association within sentences or words. Furthermore, limitations such as negations, sarcasm, ambiguity or cultural bias are making this topic difficult.

Joshi, Prabhu, Shrivastava, and Varma (2016) also focus on sentiment analysis of financial news articles, but take this research further by using the polarity of news articles as a means of predicting future trends of stocks and using this to potentially gauge the fluctuation in stock prices, by considering news articles about a company as prime information and by trying to classify news as good (positive) or bad (negative). They hypothesized that if the news sentiment is positive, there are more chances that the stock price will go up, but if the news sentiment is negative, then stock price may go down. They selected three classifiers: Random Forest (RF), Support Vector Machine and Naïve Bayes. From their results, SVM had the highest accuracy, while Naïve Bayes had the lowest accuracy of the classifiers. Additionally, the bag of words technique was used for text

mining. However, the chosen data only concerned news articles of a single company as opposed to a randomized collection of financial news articles.

Asghar, Ahmad, Marwat, and Kundi (2015) performed survey of techniques to analyse opinions posted by users about a particular video on YouTube. They utilized SVM to classify the polarity of these videos. While their study focuses on an entirely different dataset as opposed to this thesis, it does provide support for the use of SVM as a potential classifier for sentiment analysis of news articles. Additionally, Urologin (2018) uses logistic regression, RF and Adaboost as classifiers and importantly observes that logistic regression has the lowest accuracy of these classifiers. However, an important distinction in their data is that it is collected entirely from a single website (the BBC), whereas this thesis utilizes a dataset that contains financial news articles from various news sources and time periods.

Soelistio and Surendra (2015) utilizes sentiment analysis for a different application – that of identifying the polarity of public opinion toward various politicians. Importantly, they rely on the usage of Naïve Bayes classifier to determine the polarity of the news sources in their dataset and provide an overview of literature that support the use of this classifier to determine the polarity of reader's opinion for a variety of sources.

Pohan, Budi, and Suryono (2020) , in their work on determining polarity of news articles concerning P2P learning in Indonesia, rely on several models utilised in this thesis, namely; Naïve Bayes, Logistic Regression and SVM. They further confirm the opinion shared by other related work in this field – that SVM provides the highest accuracy in classifying polarity of text in sentiment analysis. They also highlight some important considerations to be taken into account while performing such an analysis. Conducting sentiment analysis on long texts has a greater challenge especially since the text is online news. Online news has a combination of more formal words and it is difficult to distinguish neutral and positive meanings, which is due to the unbalanced number of datasets for positive, neutral and negative sentiment labels. This is indicated by the interpretation of the results of the confusion matrix.

Setty et al. (2014) also follow a similar pattern in their study by using similar classifiers – logistic regression, Naïve Bayes and SVM – and echo similar findings, that SVM slightly outperforms other classifiers, but all classifiers have a high degree of accuracy. Their paper focuses on Facebook news feeds and attempts to classify the user's news feeds into various categories using machine learning classifiers to provide a better representation of user-data profiles. By automatically classifying Facebook news feeds into life posts and entertainment posts and performing sentiment analysis of life event posts, they try to provide a better model of a user's online behaviour, when accessing Facebook. Their work highlights the

importance of sentiment analysis in better understanding social networks, as well as network relationships.

While the literature discussed so far seems to indicate that the predictive power of the Naïve Bayes classifier in comparison to the other selected in this thesis is weaker, it does not imply that this classifier lacks the ability to accurately perform sentiment analysis. Baid et al. (2017) provide useful insight into this particular classifier as in their study, which compares the performance ability of Naïve Bayes, K-Nearest Neighbours and Random Forest. From their work, Naïve Bayes significantly outperforms the other classifiers in terms of accuracy, and additionally has the lowest error rate of the three. Additionally, they also provide useful applications of this research which support the claims made by this thesis regarding the scope of application of sentiment analysis techniques. For example, they suggest that Intelligent systems can be developed which can provide the users with comprehensive reviews of movies, products, services etc. without requiring the user to go through individual reviews, and thus taking autonomous decisions based on the results provided by the intelligent systems. In other research, it was shown that detecting negation scopes was improving the accuracy of the sentiment analysis models (Pröllochs, Feuerriegel, & Neumann, 2015)

Gated Recurrent Unites (GRU) are a popular form of Recurrent Neural Networks (RNN). The GRU consists of gating units that control the movement of data (information) within the unit; however, it does this without making use of any extra separate defined memory cells. GRU further calculates two important gates called update and reset gates which modulate the movement of information passing into each hidden unit. While a vast majority of literature focuses on combining various deep learning techniques to create an optimal classification model, Sachin, Tripathi, Mahajan, Aggarwal, and Nagrath (2020) 's work focuses on comparing the performance of the two most popular RNN techniques available – that of GRU and LSTM. Across all chosen datasets, the GRU performed better and achieved a higher value against each performance metric – hence it was chosen as the sole deep learning model of this thesis.

Lastly, it is also important to highlight several studies that share similar aims with this thesis. Firstly, we look at the study performed by Jain and Kaushal (2018) where a vast selection of machine learning and deep learning methods are comparatively analysed. Their results vary to some extent with respect to the studies highlighted earlier in this section, as their Naïve Bayes performs better than the SVM and from their chosen selection of deep learning methods (which does not include a GRU), the LSTM model is the best overall performing model. Finally, we also highlight Gadri, Chabira, Ould Mehieddine, and Herizi (2021) 's work where a

similar study is conducted. From the machine learning models, the logistic regression models outperforms the Naïve Bayes and SVM, whereas the best overall model is the Deep Neural Network (DNN) model created by the authors. However, an important point to note regarding this study is the DNN model was reporting 100% accuracy. These results must not been seen as an absolute certainty, but an indicator of how different models are expected to perform on the same task.

3 METHODS

3.1 *Design*

This project was fully conducted in Python 3.7 (Van Rossum & Drake, 2009), with Jupyter Notebook being the primary IDE relied upon. Packages and libraries such as NumPy(Harris et al., 2020) and Pandas(McKinney et al., 2010) were used for pre-processing the data. This was the first step in this analysis, in order to handle the data in a more simplified and easier manner. Different functions were utilized from various libraries such as Sklearn (Pedregosa et al., 2011), Keras (Chollet et al., 2015), TensorFlow (Abadi et al., 2015) and the Natural Language Toolkit (NLTK) (Bird, Klein, & Loper, 2009). These functions were user for cleaning and analysing the data, but also in ensuring that only the relevant characters and words were used for the analysis. Additionally, Matplotlib (Hunter, 2007) and Seaborn(Waskom et al., 2017) were used for the purposes of visualizing the data.

3.2 *Data collection*

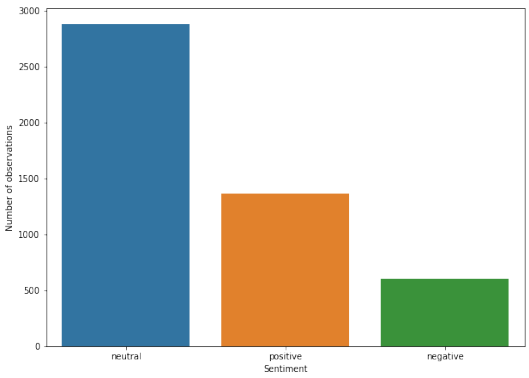
The dataset used for training the machine learning models in this project was originally introduced by Malo et al. (2013), and it can be freely found on Kaggle. This dataset was created in order to solve the problem of low utilization of statistical techniques in the financial domain, which was mostly due to the lack of good quality data for training. It contains the sentiments for financial news headlines from the perspective of a retail investor – the sentiment can be either positive, negative or neutral. As opposed to some of the narrower approaches discussed in the previous section on related work, this dataset contains English news of all listed companies from a particular stock exchange, that of the OMX in Helsinki. The news has been downloaded from the LexisNexis database using an automated web scraper.

From this database, a random subset of 10,000 articles was selected to obtain a good and general coverage across not only small and large

companies, but companies in different industries, as well as different news sources (Malo et al., 2013). Furthermore, they excluded all sentences which did not contain any of the lexicon entities. This process reduced the overall sample to 53,400 sentences, where there is at least one or more recognized lexicon entity, in each sentence. The sentences were then classified according to the types of entity sequences detected. Finally, a random sample of almost 5000 sentences was selected to represent the overall news database.

More precisely, it consists of 4850 sentences, classified into positive(1363), negative(608) and neutral(2879) groups, by considering only the information explicitly available in the given sentence. The distribution of the dataset can be seen in Figure 1. In addition, these sentences were annotated by 16 different people with adequate background in the financial markets (3 researchers and 13 master's students from Aalto University School of Business, Finland, with majors primarily in finance, accounting, and economics). Given the fact that the scope of this study was manly on the financial market, the annotators were asked to analyse the headlines from an investor point of view, more precisely weather the news article will have a positive, negative or neutral effect on the stock price of the company.

Figure 1: Dataset distribution across all three sentiments



Due to the large number of overlapping annotations (5-8 annotations per sentence), there are multiple ways to build the dataset, based on a majority vote-based gold standard. Therefore, to provide an objective comparison, four different datasets were analysed and compared for the purposes of this thesis - based on the strength of majority agreement: sentences with 100% agreement; sentences with more than 75% agreement; sentences with more than 66% agreement and sentences with more than 50% agreement (Malo et al., 2013).

However, it is important to take into account several considerations made by the creators of the selected dataset regarding their research design. They do not annotate for the following: whose opinion is at hand, or relevance of the sentence, as they assume that the both will play a smaller role towards the general sentiment of the headline. Additionally, they go on to further assume that most of the selected sentences should be relevant for a particular company on the index, as they the company name is mentioned in the article, and the articles selected originate from the financial press, so they should be concise and directly address financial incidents and developments.

3.3 *Data pre-processing*

Once the data was downloaded and subsequently ready to be extracted, with the help of Pandas library, the entire data is merged into four Dataframes that will later be utilized for the pre-processing steps. This process primarily consisted of eliminating null and duplicated values and further normalizing the data. As mentioned earlier, multiple functions were utilised for the cleaning of data The cleaning process makes sure to eliminate from the news headlines, any additional and irrelevant characters such as punctuation marks, that can potentially add noise in our data and further skew our results. Moreover, all letters were converted into lowercase, while words were compared to the English dictionary and removed in case they did not match with any word from the dictionary.

By using the various functions from the NLTK library, I was able to look at all the stop-words in the English dictionary and compare them to my data. In order to see the difference that the stop-words make into my analysis, I decided to train most of the models with the stop-words included, as well as with the stop-words removed. I decided upon this due to the fact that some models use word embeddings that usually work better with more raw data, compared to the classical ones that prefer the data as clean as possible. Moreover, techniques such as lemmatisation and stemming were used in order to group similar words with the same meaning together and to reduce all words to their root form, respectively. But following the same reasoning, some models were trained without applying these technique because this way they can capture more complex relationships.

Lastly, all words were passed through a tokenization process. This technique transforms all unique words into unique integers, based on the frequency of each word in our dictionary of words. Because the total number of words in our dictionary is close to 10.000, doing a one hot encoding will imply having a vector of length 10.000 for each word. This is

not a very efficient way to encode the data and therefore is not suitable for this analysis.

A more practical and efficient solution in this case, would be a dense encoding, meaning taking a high dimensional vector space, and assigning each word to a location in this space. In this project, the vector space chosen was 120, because values close to this number were recording the best accuracies. Each vector in this space will be of length 120, and therefore a much more memory-friendly way of encoding the words. This technique is also known as word embedding and it basically means that the location a word is sent in this high dimensional space is chosen by the model itself. A keras embedding layer is used to process this. However, because not all sentences are of equal length, a padding function from keras was used allowing us to pass all sentences and pad them to an equal length, in this case to the length of the longest sentence available in the dataset. This way, the short sentences will be filled with '0's until they reach the length of the longest sentence in the dataset.

3.4 *Implementation*

Once the data was normalized and all sentences could be passed as inputs with uniform size, the data is ready to be modelled. For this project, I chose to start with some basic, yet very popular classifiers that were also discussed in the literature; namely – SVM, NB and Logistic Regression. The goal was to test their performances on this dataset and compare the results against a more complex algorithm (i.e., a recurring neural network in the form of a GRU) and conclude which classifier is the most suitable for this task.

3.4.1 *Naive Bayes*

The Naïve Bayes algorithm is a very popular statistical model. It is a supervised learning model that is based on applying the Bayes' Theorem and always counting for the 'naïve' assumption of conditional independence between all pair of features. Although it might seem to have a simplistic architecture, the Naïve Bayes classifier is very efficient for classification problems, with wide real-life use cases, such as email spam filtering. Another advantage of this model is its speed. It requires a limited amount of training data and can work significantly faster compared to more complex algorithms. Because of its ability to estimate each distribution as an independent one-dimensional distribution, the problem of dimensionality is eliminated. However, although the Naive Bayes model usually registers

good accuracies, the probability outputs are not very accurate and therefore it is considered a relatively bad estimator.

3.4.2 *Logistic Regression*

Logistic regression is a machine learning algorithm often used for classification problems and predictive analytics. It works by assigning a probability to an event, based on an already given dataset of independent variables. Since the outcome is interpreted as a probability, the set of values that the dependent variable can reach is bounded between 0 and 1. The log likelihood function is produced over many iterations and the logistic regression aims to find the best parameter and return a predicted probability.

3.4.3 *Support Vector Machine*

Despise being considered one of the simplest models, Support Vector Machines (Cortes and Vapnik 1995) seem to be very powerful when dealing with classification problems. SVMs tend to be very effective when having to work with high dimensional data and can be considered to some level memory-efficient, as they work with subsets of the training points (that are called support vectors) when making a decision. As can be evidenced in the related work section, SVM is a popular method of choice while addressing sentiment analysis tasks and has a level of accuracy similar to that shown by using other machine learning models.

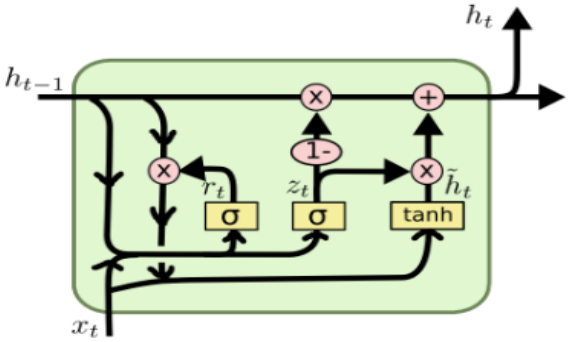
3.4.4 *GRU*

Lastly, a Recurrent neural network in the form of a GRU was modelled. RNN is a method that is used to learn long term dependencies between the features. In this case, the difference between a normal neural network and an RNN is that the normal neural network would take each sentence and collect 50 features, each feature representing a word in a given place in the sentence. However, because all features will be represented as a location of a word in a sentence, these feature will become independent on each other and therefore, there is no way for the model to understand the relationship between the features. However, a RNN looks at each feature one by one while also taking into consideration the past words that have come up before. Basically, at each time stamp it takes into consideration both the current word and a vector that represent the previous history of the sentence. This way, in the end the output should have some valuable information extracted from the sentence.

In particular, a GRU is a type of RNN, that is very similar to the Long short term memory (LSTM). It usually performs better on smaller datasets, like the one collected for this project. What makes LSTMs and

GRUs very powerful for this type of tasks are the following features. They consist of constant loop that at each timestamp, a part of the information if forgotten(forget gate), as can be seen in Figure 2. A sigmoid function is applied to the sentence vector coming through and due to the nature of the the sigmoid function(the set of values is between 0 and 1), when multiplying pointwise on the cell state, it has the effect of forgetting some of the information in that cell e.g. you can either multiply by 0, which translates to forget everything, or multiply by 1, which translates to remember everything, or multiply with any value in between that would result in a proportion of the original information being lost. Therefore, at each iteration, parts of the cell state are forgotten, while at the same time adding new information taken from the hidden state. At the very end, the output is slightly modified by a 'Tanh' function with a sigmoid activation.

Figure 2: GRU structure



For this project, a GRU with a dense layer of 256 neurons was modelled. Because I will work with four datasets and compare the results between them, the parameters of the network are slightly different for each dataset, due to the fact the input size must be adjusted accordingly. The activation function used for this model is a 'tanh' activation function, which seemed to outperform the 'relu' activation function in both speed and efficiency. The network also needs an output layer, with 3 neurons, one for each class and it uses a 'softmax' activation function. This function will assign to each class a value between 0 and 1, which can be interpreted as the probability values for each class (they will all sum in total to 1, forming the total probability).

The model will be compiled using an 'adam' optimized and a 'sparse categorical crossentrompy' as the loss function, since it is suitable for a multiclass classification problem The 'return_sequences' parameter of the GRU model is by default set to 'False'. Changing it to 'True', means predicting at every timestamp a temporary output and returning a two-

dimensional array of the outputs from each timestamp, and therefore capturing much more information this way. In particular, this adjustment proved to improve the performance of the model. However, once we set the parameter to 'True' the output becomes 2 -dimensional and a flatten layer needs to be added that transforms the output back to 1 dimension.

3.5 *Evaluation*

For the purposes of evaluation, accuracy is the primary metric used for comparing the machine learning (ML) models and the more complex deep learning model, across the various datasets selected. Additionally, for the ML models, multiple metrics will be relied upon to make an intra-group comparison to evaluate their performance on all four datasets. Additionally, because the dataset used for this project was already annotated by 16 human judges with adequate background, this data will serve as the ground truth labelled data that will be used for computing the above metrics.

Additionally, it is important to point out that although a random guess should normally give accuracies of around 30% for a 3-class classification task, because in this case the dataset is not uniformly distributed across the sentiments, a base line model that will always predict the 'neutral' class is registering accuracies of 59%.The results ca be found in the following section.

4 RESULTS

In this section, an overview of the main finding will be presented, together with tables and figures for clearly visualising and understanding the numbers behind the results.

The experimental study involved the classification of news headlines collected from various companies, using different machine learning classifiers and a recurrent neural network in the form of a GRU. All models were tested and compared across four datasets. These datasets differ in the level of agreement between annotators (50%, 66%, 75%, 100%), and therefore also differ in size (e.g. the 100% agreement dataset has the least observations, as it only includes sentences which all judges have annotated in the same way).

The first part of the analysis involves the main dataset of the project, the "50% agreement dataset" as it provides a larger number of observations for training the models. The performances of the discussed classifiers can be found in Table 1, where the accuracy can be found at the top in bold font, while below it, metrics such as precision, recall and f1-score across all three

classes are presented. From this table, the highest accuracy is registered by the Logistic Regression model, with a maximum of 77%, while the Naïve Bayes and the Support Vector Machine have similar, but slightly lower accuracies of around 73%.

Table 1: Results for the 50%agreement dataset

	Naive Bayes	Logistic Regression	Support Vector Machine
accuracy	**0.73**	**0.77**	**0.73**
precision	negative: 0.58 neutral: 0.77 positive: 0.70	negative: 0.79 neutral: 0.78 positive: 0.73	negative: 0.96 neutral: 0.71 positive: 0.80
recall	negative: 0.61 neutral: 0.87 positive: 0.52	negative: 0.53 neutral: 0.89 positive: 0.62	negative: 0.31 neutral: 0.98 positive: 0.41
f1-score	negative: 0.59 neutral: 0.82 positive: 0.60	negative: 0.64 neutral: 0.83 positive: 0.67	negative: 0.47 neutral: 0.82 positive: 0.54

Secondly, the dataset is changed from '50% agreement' to '66% agreement'. Although the number of observations is lower and the training size is consequently lower too, there is a higher level of agreement between the annotators and therefore the data might be more accurate. Table 2 provides the results of this analysis. The logistic regression model is again performing the best, with a maximum accuracy of 81%. The Naïve Bayes classifiers registers 77% while the Support Vector machine only 76%.

Table 2: Results for the 66%agreement dataset

	Naive Bayes	Logistic Regression	Support Vector Machine
accuracy	**0.77**	**0.81**	**0.76**
precision	negative: 0.65 neutral: 0.79 positive: 0.77	negative: 0.81 neutral: 0.82 positive: 0.80	negative: 0.85 neutral: 0.74 positive: 0.80
recall	negative: 0.66 negative: 0.89 positive: 0.57	negative: 0.62 neutral: 0.92 positive: 0.67	negative: 0.35 neutral: 0.98 positive: 0.46
f1-score	negative: 0.66 neutral: 0.84 positive: 0.62	negative: 0.70 neutral: 0.87 positive: 0.73	negative: 0.50 neutral: 0.84 positive: 0.59

Thirdly, the dataset is changed once again, from the '60% agreement' to the '75% agreement' dataset. The performances of the models can be seen in Table 3. As it was the case with the last experiment, the accuracies

increase across all classifiers. Both the Naïve Bayes and the Support Vector Machine seem to record similar accuracies, with a maximum of 82%, while the Logistic Regression outperforms both of them, with a maximum accuracy of 86%.

Table 3: Results for the 75%agreement dataset

	Naive Bayes	Logistic Regression	Support Vector Machine
accuracy	**0.82**	**0.86**	**0.82**
precision	negative: 0.69 neutral: 0.86 positive: 0.79	negative: 0.86 neutral: 0.87 positive: 0.84	negative: 0.89 neutral: 0.81 positive: 0.87
recall	negative: 0.72 negative: 0.90 positive: 0.67	negative: 0.71 neutral: 0.95 positive: 0.72	negative: 0.49 neutral: 0.99 positive: 0.57
f1-score	negative: 0.70 neutral: 0.88 positive: 0.72	negative: 0.78 neutral: 0.91 positive: 0.78	negative: 0.63 neutral: 0.89 positive: 0.69

Moreover, the last dataset used for testing the performance of the presented machine learning classifiers was the '100% agreement dataset'. It consists of almost half the observations available in the main dataset, but only includes sentences annotated the same way by all 16 annotators, and therefore it has arguably the most accurate data. As expected, the results recorded in this experiment are higher, compared to the previous datasets and can be seen in Table 4. The Logistic Regression seems to outperform the other classifiers, with an accuracy of 89%. The same pattern can be seen here as well: both the Naive Bayes and the Support Vector Machine classifiers record better accuracies compared to the previous experiments, but still slightly lower compared to the Logistic Regression, with a maximum of around 84%.

Lastly, the performance of the GRU model, across all datasets can be seen in Table 4. It follows the same behaviour of increased accuracy as the level of agreement is also increasing. The results recorded seem to be higher compared to both the Naïve Bayes and the Support Vector Machine across all datasets, but just under the performance of the Logistic Regression.

5 DISCUSSION

As evidenced from the results of our study, the logistic regression model is the best performing model out of those that have been selected, with a model accuracy ranging between 77-89%. However, the other machine

Table 4: Results for the 100%agreement dataset

	Naive Bayes	Logistic Regression	Support Vector Machine
accuracy	**0.84**	**0.89**	**0.84**
precision	negative: 0.67 neutral: 0.90 positive: 0.74	negative: 0.86 neutral: 0.90 positive: 0.86	negative: 0.83 neutral: 0.86 positive: 0.78
recall	negative: 0.62 negative: 0.93 positive: 0.69	negative: 0.68 neutral: 0.98 positive: 0.74	negative: 0.46 neutral: 0.99 positive: 0.61
f1-score	negative: 0.64 neutral: 0.91 positive: 0.71	negative: 0.76 neutral: 0.94 positive: 0.80	negative: 0.59 neutral: 0.92 positive: 0.69

Table 5: Results for the GRU model across all datasets

	50% agreement	66% agreement	75% agreement	100% agreement
Test loss	63.015%	60.949%	57.315%	49.509%
Test accuracy	**76.320%**	**76.967%**	**79.745%**	**84.982%**

learning models do not perform as well as the logistic regression , tallying accuracies ranging from 73%-84%. However, it is important to note that while the Naïve Bayes appears to marginally perform better than the support vector machine, this slight performance increase is not recreated in every iteration of our study. Only when the percentage of agreeing annotators is at the lower ends do we see an extremely slight increase in performance in the Naïve Bayes model.

Additionally, with respect to our selected deep learning model (that of a RNN in the form of a GRU), it is important to note that it consistently outperforms both the Naïve Bayes and Support Vector Machine models. However, and more importantly, it is not the best performing model and its performance can be ranked just under that of the Logistic Regression model. While there are multiple studies that highlight the predictive power of a logistic regression model in sentiment analysis, it is important to note that there is a lack of literature focusing on its comparison with a GRU.

Another important talking point is the dataset itself. There appears to be a relationship between the accuracy of the selected model and proportion of annotators that agree on the sentiment of articles compiled in the selected dataset. This relationship can be observed across all the selected models, as there is a collective increase in accuracy that is directly proportionate to the percentage of annotators that agree on the sentiment of the news in the dataset. The higher the percentage of agreement amongst annotators,

the smaller the dataset becomes. This is because the annotators with their domain knowledge and expertise do not universally agree of the polarity of sentiment attached to all sentences in the dataset. Therefore, as highlighted in the results, the accuracy of the selected models increases with an increase in the percentage of agreeableness between annotators.

5.1 *Limitations*

There are two notable limitations in this thesis. Firstly, from the available supporting literature and related work on the comparative performance of deep learning versus machine learning models on tasks of sentiment analysis, there are no studies that perform a comparison between the exact same selection of models as reflected in this thesis. This can create a problem as there is a lack of comparable literature to serve as a baseline, against which the results of this thesis can be compared with. Additionally, many of the related studies perform such a comparative analysis with the ultimate aim of building an optimal classifier that can outperform any conventionally used techniques. For example, important features of LSTM and GRU are combined to generate ensemble models which are then comparatively analysed against machine learning models.

Secondly, the dataset itself serves as a limitation while performing such a study. This is because there is almost no similarity between the datasets utilized by related studies in performing comparative sentiment analysis. Furthermore, in relation to the dataset selected for this thesis, there are no similar studies that evaluate the polarity of the contents of the dataset via an additional process involving annotators with domain expertise.

6 CONCLUSION

The primary aim of this thesis was to comparatively analyse the performance of machine learning and deep learning models, in the task of sentiment analysis. Logistic Regression, Support Vector Machine and Naïve Bayes were selected as our machine learning models, whose performances were compared with a single deep learning model – that of a RNN utilizing a GRU. Our chosen dataset was a collection of English news articles addressing the various companies listed on a stock exchanged based out of Helsinki. This dataset was further supported by the analysis of multiple annotators, who also attempted to gauge the polarity of news articles in the dataset by relying on their domain expertise in the field of finance. From a pre-processing perspective, the dataset was split into multiple smaller datasets ranked in order of the percentage of agreeableness exhibited between the various annotators. This implies that the datasets with

the highest percentage of agreeableness (i.e., the dataset of articles which the annotators assigned the highest proportion of similar ratings) contain fewer articles (observations).

Upon performing our study, the model with the highest accuracy was the logistic regression model, which even outperformed our deep learning model (the GRU). However, the GRU was not significantly worse vis-à-vis accuracy, but consistently finished behind the logistic regression model. Additionally, the GRU was well ahead of the other machine learning models selected, namely; the SVM and NB models. Another important point to note is that the accuracy of models was higher in the datasets with a higher percentage of agreeableness between the annotators, irrespective of the nature of the model (i.e., for both machine learning and deep learning models).

These results appear to slightly contradict the findings of previous similar studies, which almost always place deep learning models ahead of machine learning models. However, it is important to note that there is a scarcity in the research regarding the direct comparison of Logistic Regression versus GRU for a task of sentiment analysis. This could serve as an important avenue for future research. Another important aspect of this thesis which should be researched further is the impact of annotators in the analysis of model performance. This study indicates that model accuracy could be proportional to the percentage of agreeableness between annotators who have independently attempted to classify the polarity of the textual data used in performing sentiment analysis.

REFERENCES

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., ... Zheng, X. (2015). *TensorFlow: Large-scale machine learning on heterogeneous systems.* Retrieved from https://www.tensorflow.org/ (Software available from tensorflow.org)

Asghar, M. Z., Ahmad, S., Marwat, A., & Kundi, F. M. (2015). Sentiment analysis on youtube: A brief survey. *arXiv preprint arXiv:1511.09142*.

Baid, P., Gupta, A., & Chaplot, N. (2017, 12). Sentiment analysis of movie reviews using machine learning techniques. *International Journal of Computer Applications*, *179*, 45-49. doi: 10.5120/ijca2017916005

Bird, S., Klein, E., & Loper, E. (2009). *Natural language processing with python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Cambria, E., Schuller, B., Liu, B., Wang, H., & Havasi, C. (2013). Knowledge-based approaches to concept-level sentiment analysis. *IEEE intelligent systems*, *28*(2), 12–14.

Chollet, F., et al. (2015). *Keras.* GitHub. Retrieved from https://github

.com/fchollet/keras

Gadri, S., Chabira, S., Ould Mehieddine, S., & Herizi, K. (2021). Sentiment analysis: Developing an efficient model based on machine learning and deep learning approaches. In *International conference on intelligent computing & optimization* (pp. 237–247).

Godbole, N., Srinivasaiah, M., & Skiena, S. (2007, 01). Large-scale sentiment analysis for news and blogs..

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., ... Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, *585*, 357–362. doi: 10.1038/s41586-020-2649-2

Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. *Computing in science & engineering*, *9*(3), 90–95.

Jain, K., & Kaushal, S. (2018). A comparative study of machine learning and deep learning techniques for sentiment analysis. In *2018 7th international conference on reliability, infocom technologies and optimization (trends and future directions)(icrito)* (pp. 483–487).

Joshi, A., Prabhu, A., Shrivastava, M., & Varma, V. (2016). Towards sub-word level compositions for sentiment analysis of hindi-english code mixed text. In *Proceedings of coling 2016, the 26th international conference on computational linguistics: Technical papers* (pp. 2482–2491).

Kirange, M. D. K., & Deshmukh, D. R. R. (2016, Mar.). Sentiment analysis of news headlines for stock price prediction. *COMPUSOFT: An International Journal of Advanced Computer Technology*, *5*(3). Retrieved from https://ijact.in/index.php/ijact/article/view/473

Kogan, S., Moskowitz, T. J., & Niessner, M. (2019). Fake news: Evidence from financial markets. *Available at SSRN*, *3237763*.

Malo, P., Sinha, A., Takala, P., Korhonen, P., & Wallenius, J. (2013, July). *Good Debt or Bad Debt: Detecting Semantic Orientations in Economic Texts* (Papers No. 1307.5336). arXiv.org. Retrieved from https://ideas.repec.org/p/arx/papers/1307.5336.html

Märkle-Huß, J., Feuerriegel, S., & Prendinger, H. (2017). Improving sentiment analysis with document-level semantic relationships from rhetoric discourse structures. In *Proceedings of the 50th hawaii international conference on system sciences* (pp. 1142–1151).

McKinney, W., et al. (2010). Data structures for statistical computing in python. In *Proceedings of the 9th python in science conference* (Vol. 445, pp. 51–56).

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... others (2011). Scikit-learn: Machine learning in python. *Journal of machine learning research*, *12*(Oct), 2825–2830.

Pohan, N. W. A., Budi, I., & Suryono, R. R. (2020). Borrower sentiment on p2p lending in indonesia based on google playstore reviews.

In *Proceedings of the sriwijaya international conference on information technology and its applications (siconian 2019)* (p. 17-23). Atlantis Press. Retrieved from https://doi.org/10.2991/aisr.k.200424.003 doi: https://doi.org/10.2991/aisr.k.200424.003

Pröllochs, N., Feuerriegel, S., & Neumann, D. (2015). Generating domain-specific dictionaries using bayesian learning. In *Ecis*.

Sachin, S., Tripathi, A., Mahajan, N., Aggarwal, S., & Nagrath, P. (2020). Sentiment analysis using gated recurrent neural networks. *SN Computer Science*, *1*(2), 1–13.

Setty, N. K. H., Nagaraja, M. S., Nagappa, D. H., Giriyaiah, C. S., Gowda, N. R., & Naik, R. D. M. L. (2014). A study on surgical site infections (ssi) and associated factors in a government tertiary care teaching hospital in mysore, karnataka. *International Journal of Medicine and Public Health*, *4*(2).

Soelistio, Y. E., & Surendra, M. R. S. (2015). Simple text mining for sentiment analysis of political figure using naive bayes classifier method. *arXiv preprint arXiv:1508.05163*.

Urologin, S. (2018). Sentiment analysis, visualization and classification of summarized news articles: a novel approach. *International Journal of Advanced Computer Science and Applications*, *9*(8).

Van Rossum, G., & Drake, F. L. (2009). *Python 3 reference manual*. Scotts Valley, CA: CreateSpace.

Wan, X., Yang, J., Marinov, S., Calliess, J.-P., Zohren, S., & Dong, X. (2021, 02). Sentiment diffusion in financial news networks and associated market movements. *Scientific Reports*, *11*. doi: 10.1038/s41598-021-82338-6

Waskom, M., Botvinnik, O., O'Kane, D., Hobson, P., Lukauskas, S., Gemperline, D. C., ... Qalieh, A. (2017, September). *mwaskom/seaborn: v0.8.1 (september 2017)*. Zenodo. Retrieved from https://doi.org/10.5281/zenodo.883859 doi: 10.5281/zenodo.883859

Yadav, A., Jha, C., Sharan, A., & Vaish, V. (2020, 01). Sentiment analysis of financial news using unsupervised approach. *Procedia Computer Science*, *167*, 589-598. doi: 10.1016/j.procs.2020.03.325