Master of Science in Computer Science
January 2023

Deep Learning Based Sentiment Analysis

Shashank Kalluri

Faculty of Faculty, Blekinge Institute of Technology, 371 79 Karlskrona, Sweden

This thesis is submitted to the Faculty of Computing at Blekinge Institute of Technology in partial fulfilment of the requirements for the degree of Master of Science in Computer Science. The thesis is equivalent to 20 weeks of full time studies.

The authors declare that they are the sole authors of this thesis and that they have not used any sources other than those listed in the bibliography and identified as references. They further declare that they have not submitted this thesis at any other institution to obtain a degree.

Contact Information:
Author(s):
Shashank Kalluri
E-mail: shka21@student.bth.se

University advisor:
Hüseyin Kusetogullari
Department of Computer Science

Faculty of Faculty
Blekinge Institute of Technology
SE–371 79 Karlskrona, Sweden

Internet : www.bth.se
Phone : +46 455 38 50 00
Fax : +46 455 38 50 57

Abstract

Background: Text data includes things like customer reviews and complaints, tweets from social media platforms. When analyzing text-based data, the Sentiment Model is used. Understanding news headlines, blogs, the stock market, political debates, and film reviews some of the areas where sentiment analysis is used. The results of a sentiment analysis may be used to aid in evaluating whether a review is favorable, negative, or neutral. In this thesis we explore the performance of some algorithms.

Objectives: The problems with natural language processing, on the other hand, make it harder for sentiment analysis to work well and be accurate (NLP). In the past few years, it has been shown that deep learning models are a promising way to solve some of NLP's problems. This paper looks at the most recent studies that used deep learning to solve problems with sentiment analysis and their performance metrics

Methods: The literature review is done to figure out which algorithms are best for achieving the above goals. An experiment is done to understand how deep learning works and what metrics are used to figure out which model is the best for sentiment analysis. Several datasets have been used to test models that use the term frequency-inverse document frequency and word embedding.

Results: The experiment indicated that the CNN model strikes the best balance between how fast it works and how well it works. When used with word embedding, the RNN model was the most accurate, but it took a long time to process and didn't work well with TF-IDF. The processing times and results of DNN are about average.

Conclusions: The primary objective of this research is to learn more about the fundamentals of deep learning models and related approaches that have been used for sentiment analysis of social network data. Before feeding it to deep learning models, we changed the data using TF-IDF and word embedding. Architectures for DNN, CNN, and RNN were looked into after performing the literature review. The processing time gap was fixed, and the best combination was found.

Keywords: Sentiment Analysis, Word Embedding, Deep Learning,

Acknowledgments

I'd want to convey my heartfelt appreciation to my supervisor, Christina Jenkins, for providing me with the incredible chance to work at Devoteam Creative Tech. Throughout the thesis, they have been really patient and helpful in sharing their knowledge.

I would like to express my gratitude to my University supervisor Dr.Hüseyin Kusetogullari for his constant support and for helping me throughout the thesis.It was a great privilege and honor to work under his supervision

I finally thank my family members and friends who supported me all the time including the wonderful people I have met during my thesis.

Contents

Abstract i

Acknowledgments iii

1 Introduction 1
- 1.1 Problem statement . . . 2
- 1.2 Aim and Objectives . . . 2
  - 1.2.1 Aim . . . 2
  - 1.2.2 Objectives . . . 2
- 1.3 Research Questions . . . 3
- 1.4 Outline . . . 3

2 Background 5
- 2.1 Machine Learning . . . 5
- 2.2 Traditional sentiment classification techniques . . . 5
- 2.3 Deep Learning . . . 6
  - 2.3.1 Deep Neural Network . . . 7
  - 2.3.2 Convolutional Neural Networks (CNN) . . . 8
  - 2.3.3 Recurrent Neural Networks (RNN) . . . 9
- 2.4 Sentiment Analysis . . . 10
- 2.5 BERT . . . 12
- 2.6 Performance Measures . . . 12
  - 2.6.1 Accuracy . . . 13
  - 2.6.2 Precision . . . 13
  - 2.6.3 Recall . . . 13
  - 2.6.4 F1 Score . . . 14

3 Related Work 15

4 Method 17
- 4.1 Literature Review . . . 17
- 4.2 Experiment . . . 19
  - 4.2.1 Software Tools . . . 19
  - 4.2.2 Data . . . 20
  - 4.2.3 Dataset . . . 22
  - 4.2.4 Data Preparation . . . 23
  - 4.2.5 Training, Validation, and Test sets . . . 25
  - 4.2.6 Data Cleaning . . . 25

4.2.7 Word Embedding 26
4.2.8 TF-IDF 26
4.2.9 Implementation 26

5 Results and Analysis **29**
5.1 Systematic Literature Review Results 29
5.1.1 SLR Analysis 32
5.2 Experiment 1 - Results 32
5.3 Experiment 2 - Results 38
5.4 Observations 39

6 Discussion **41**
6.1 Threats to Validity 42
6.1.1 Internal Validity 42
6.1.2 External Validity 42
6.1.3 Conclusion Validity 42

7 Conclusions and Future Work **43**
7.1 Conclusion 43
7.2 Future Work 44

References **45**

List of Figures

2.1 Sentiment Analysis using Machine Learning 7
2.2 Sentiment Analysis using Deep Learning 7
2.3 Deep Neural Network 8
2.4 Convolutional Neural Network 9
2.5 Recurrent Neural Network 9
2.6 Categorisation of Sentiment Analysis Techniques 11
2.7 BERT Architecture 12

4.1 Web Scraping of the Website 'www.reviews.io' done using beautiful-soup library 21
4.2 This is the sample Dataset that is formed after the web scraping one of the social media channel. 22
4.3 Sample of Sitejabber Dataset 23
4.4 BERT Model 24
4.5 A sample of the labelled dataset generated 24

5.1 Accuracy values of the models with TF-IDF and WordEmbedding . . 32
5.2 Recall values of the models with TF-IDF and WordEmbedding 33
5.3 Precision values of the models with TF-IDF and WordEmbedding . . 33
5.4 F-Score values of the models with TF-IDF and WordEmbedding . . . 34
5.5 AUC values of the models with TF-IDF and WordEmbedding 34
5.6 Performance measures of the Recurrent Neural Network with each of the Word Embedding Technique 35
5.7 CPU Processing time for the various datasets 38

List of Tables

4.1 Hardware Environment. 19
4.2 Twitter Data Set: Sizes of each subset 25

5.1 SLR Results 29
5.2 Twitter Dataset 35
5.3 SiteJabber Dataset 36
5.4 Reddit Dataset 36
5.5 Review IO Dataset 36
5.6 Consumer Affairs Dataset 36

Chapter 1

Introduction

Since we often rely our choices on the experiences and perspectives of others, feelings and viewpoints play a significant part in human behaviour. Learning about someone else's experiences and ideas is helpful in gaining a more well-rounded perspective since everyone's perspectives are subjective and influenced by their own unique circumstances. There has been an interest in public opinion to institutions, businesses, and political figures over a significant length of time. The opinions of a population as a whole may be used to develop information about future subjects and trends, as well as provide insight into who will win elections. In addition to this, it helps assess public opinion on goods and services, which in turn assists marketing teams in deciding on a marketing plan, enhancing current or manufacture of a new product, and increasing customer assistance [1, 2, 6] As a result, the acknowledgment of sentiment in a variety of professions is of the utmost importance.

At first, people's thoughts were gathered and analysed by hand after being gathered via the use of questionnaires and surveys. On the other hand, as people's familiarity with the internet grew, they began posting their thoughts and behaviors on the web more regularly [1]. The proliferation of social media platforms in recent years has made it possible for users of these platforms to disseminate a broad range of information and to offer a greater number of methods in which they may express their thoughts [2]. Blogs, discussion forums, reviews, comments, and microblogging services like Twitter and Facebook all serve as valuable data sources since they include audio recordings, video files, picture files, and opinions. Other rich data sources include reviews and comments. [3, 4]

This paper looks at how people feel and think about a furniture store's new product based on online reviews of it. Especially how much attention it gets and how many good and bad feelings it makes people feel. Researchers have come up with a lot of strategies and algorithms for figuring out how people feel about something. The analysis of this kind analyzed assists businesses in better understanding their customers' attitudes regarding their brand efforts. Sentiment Analysis is a kind of Natural Language Processing that makes use of a variety of techniques - Machine Learning algorithms, Lexicon based algorithms and Hybrid algorithms to classify data [7, 9].In the past few years, a number of studies have come up with ideas for deep-learning-based sentiment analyses. These analyses have different features and

levels of performance. This work looks at the most recent studies that used deep learning models to solve different problems related to sentiment analysis. [5]

1.1 Problem statement

In the past few years, a number of studies have come up with ideas for deep-learning-based sentiment analyses. [5] These analyses have different features and levels of performance. This work looks at the most recent studies that used deep learning models to solve different problems related to sentiment analysis.We applied deep learning models with TF-IDF and word embedding to Twitter datasets and implemented the state-of-the-art of sentiment analysis approaches based on deep learning.

1.2 Aim and Objectives

1.2.1 Aim

This thesis aims to explore the different combinations of application of the deep learning techniques and publish their comparative study, performed on the data available on the social media networks for a furniture store and derive insights from it. Most papers that do comparison studies focus on reliability metrics like overall accuracy or F-score and ignore processing time out. This thesis addresses that gap. Also, only a small number of datasets are used to evaluate the models.

1.2.2 Objectives

- To apply different word embedding methods with deep learning techniques.
- To find the most popular deep learning Techniques.
- Discussion on the processing time of the deep learning models.

1.3 Research Questions

RQ1: Which are the popular deep learning techniques used to perform sentiment analysis?

Justification: By doing a thorough literature study, the research issue is addressed (SLR). The Sentiment Analysis algorithm is executed by a collection of algorithms. The SLR enables us to examine prior research and locate popular methods for our thesis.

RQ2: Which combination of word embedding performs the best with the deep learning model?

Justification: This research question contributes to the comparative study of different word embedding techniques in combinations with techniques obtained from RQ1.The Experimentation method is used to solve this question.

RQ3:Which Deep Learning model has the best processing time?

Justification: The need for RQ3 is to address the gap of the processing time of different models.The Experimentation method is used to solve this question.This questions is used to help in determing the computing cost as well.

1.4 Outline

This section describes the thesis structure

Chapter 1: The Introduction and motivation of the thesis, as well as the aim and is an overview of the thesis and the problem that we are trying to solve. It talks about the purpose, goals, and research questions.

Chapter 2: This chapter provides an overview of the research's technical background as well as its core principles.

Chapter 3: The methodology section gives a summary of the many algorithms that were used as a direct consequence of the literature review and the recommended strategy.

Chapter 4: The experiments that were carried out in order to address the research questions are the primary emphasis of this chapter.

Chapter 5: This chapter is where the findings from the experiment are presented.

Chapter 6: This section contains a discussion and analysis of the acquired results.

Chapter 7: This chapter provides conclusion and future work of the thesis.

Chapter 2

Background

2.1 Machine Learning

Machine learning, as the name implies, is the process of computers learning without explicit human programming. First, give them excellent data, then train them by developing several machine learning models utilising the data and different techniques. Primarily divided into two types: [3, 29]

- **Supervised Learning:** Using labelled data, supervised learning algorithms are taught.The outcome is predicted using a supervised learning model.In supervised learning, the model is fed input and output.
- **Unsupervised Learning:** Algorithms for unsupervised learning are taught on unlabeled data. Unsupervised learning models uncover data's hidden patterns. In unsupervised learning, the model is fed simply input data. Ex:- Clustering
- **Reinforcement Learning:** Reinforcement Learning (RL) is a machine learning technique that enables an agent to learn in an interactive environment by trial and error using feedback from its actions and experiences. The investigation's target is finding an appropriate action model that would maximize the agent's overall cumulative reward. RL model performs learning based on the suitable action that would maximize the agent's total cumulative reward.

2.2 Traditional sentiment classification techniques

Traditionally, supervised machine learning methods like Naive Bayes (NB), Support Vector Machine (SVM), or Logistic Regression (LR) have been used in an effort to solve the text sentiment categorization issue [22]. Pang et al. [10] produced one of the first studies to suggest that machine learning may be utilized for the categorization of content on online platforms based on the sentiment of the text. On the IMDb movie review dataset, Pang et al. compared the performance of the NB, Maximum Entropy (ME), and SVM classifiers [17, 18]. SVM was able to achieve an accuracy that was about 83 Percent, which was considered to be good. Since that time, the

use of social media has steadily increased over the years, and these days, millions of individuals express their thoughts and ideas via online platforms. An interest in attaining automated sentiment categorization has been sparked as a result of the vast amounts of emotive data that are made accessible on the majority of social network sites. [29]

Twitter has been one of the social media platforms that has been investigated the most in terms of sentiment analysis up to this point. Twitter gives users the ability to communicate their thoughts in the form of brief messages known as tweets. Users are compelled to organize their ideas in a way that is succinct but gets to to the point since the available space is restricted. This results in data that is rich in sentiment, making it suited for use in NLP activities. Neethu and Rajasree investigated and compared the effectiveness of SVM, [28] NB, [27]and ME algorithms on electronic product tweets categorization, reaching 93 Percent accuracy with SVM and ME. Agarwal et al. obtained 75 percent accuracy using SVM for binary classification on non-domain specific data. [11] [12]. There has also been some work done with YouTube datasets, such as the classification of YouTube cooking videos using SVM (with an accuracy of 95.3 Percent achieved) [19] or the classification of popular Arabic YouTube videos using their comments, with an F1-score of 0.88 achieved by SVM with the Radial Bases function [12, 16].

The classic machine learning algorithms perform poorly with cross-lingual or cross-domain data [15] and have been under performing in contrast to deep learning. Despite the fact that these techniques may yield accurate sentiment prediction for text, they also have drawbacks.

2.3 Deep Learning

By incorporating a multi layer structure into the neural network's hidden layers, deep learning is able to achieve more complex results. Features in conventional machine learning methods are specified and retrieved by hand or via the use of feature selection techniques. Deep learning models, on the other hand, automatically learn and extract information, leading to improved accuracy and performance. Classifier models' hyper parameters are often measured automatically as well. Comparison of standard machine learning (Support Vector Machine (SVM), Bayesian networks, and decision trees) with deep learning for sentiment polarity categorization is shown in Figure 2.1 and 2.2. When it comes to solving difficult issues in areas like image and voice recognition and NLP, the state-of-the-art solutions are those that use artificial neural networks and deep learning. In this part, we'll go through a variety of deep learning approaches.

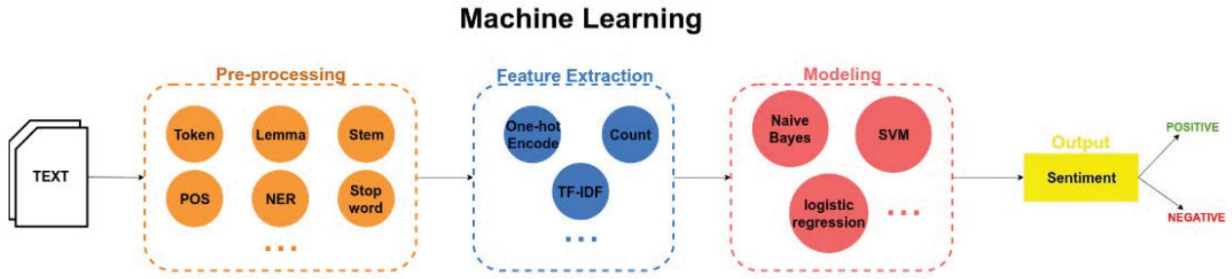

Figure 2.1: Sentiment Analysis using Machine Learning

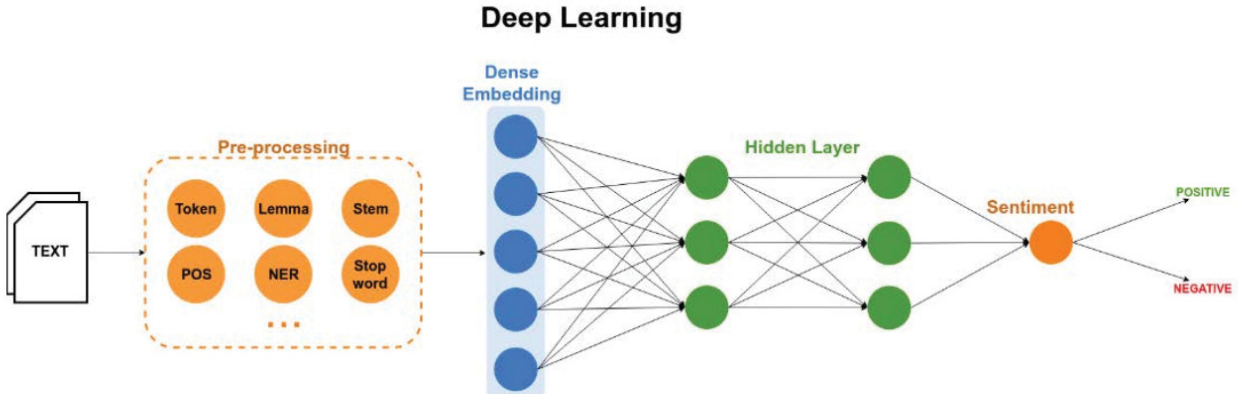

Figure 2.2: Sentiment Analysis using Deep Learning

2.3.1 Deep Neural Network

It is a new generation of machine learning [10] that mimics the structure and function of the human brain. This algorithm's distinctive characteristic enables it to automatically grasp the needed features. The deep learning model is a mathematical function f: X Y. Deep learning is the development of an ANN that employs more than one hidden layer to model a dataset [6].As shown in figure 2.3 It has three primary layers:

- Input Layer: neurons receive input from variable X.
- It contains neurons that receive signals from the preceding input layer. Each buried layer trains its own set of characteristics.The more buried layers, the more intricate abstract.
- Output Layer: This layer is made up of neurons that receive input from the hidden layer and create the output value.

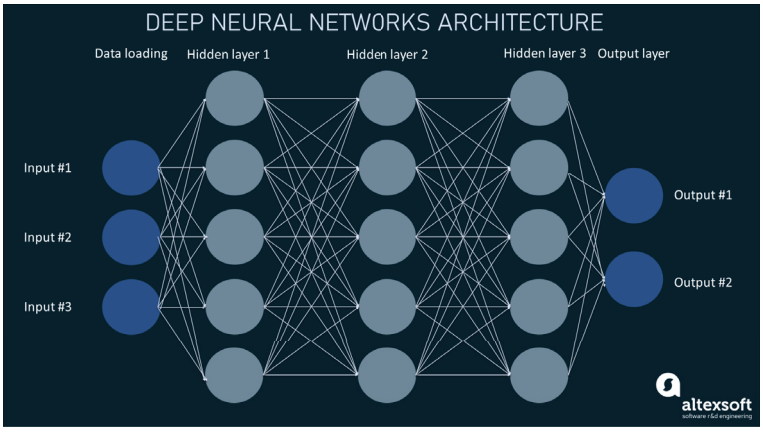

Figure 2.3: Deep Neural Network

2.3.2 Convolutional Neural Networks (CNN)

CNNs consist of multiple layers of convolutions with nonlinear activation functions, such as ReLU or tanh, applied to the results. In a conventional feedforward neural network, each input neuron is connected to each output neuron in the next layer. This is also known as a fully connected or affine layer. [19]

In CNNs, the output is computed using convolutions over the input layer. This produces local connections in which each input area is linked to a neuron in the output. Each layer applies several filters, generally hundreds or thousands as seen above, and mixes the resulting images. A CNN automatically learns the values of its filters based on the desired task during the training phase.

For instance, a CNN for image classification may learn to detect edges from raw pixels in the first layer, then use the edges to detect simple shapes in the second layer, and finally use these simple shapes to deter higher-level features, such as facial shapes, in higher layers. The last layer is a classifier that employs these high-level characteristics. Instead of picture pixels, the input to the majority of NLP jobs is a matrix of phrases or texts. Each row of the matrix represents one token, which is often a word but might also be a character. In other words, each row is a vector representing a word. These vectors are often word embeddings (low-dimensional representations) such as word2vec or GloVe, but they may also be one-hot vectors that index the word into a dictionary.

Data were preprocessed for the embedding matrix. Figure 2.4 depicts 4 convolution layers and 2 max pooling layers processing an input embedding matrix. That the very first 2 convolution layers utilize 64 and 32 filters to train various features; a max pooling layer reduces output complexity and prevents over fitting. 3 and 4 convolution layers feature 16 and 8 filters, followed by max pooling. The last layer is a fully linked layer that reduces the 8-dimensional vector to a 1-dimensional output vector (Positive, Negative)

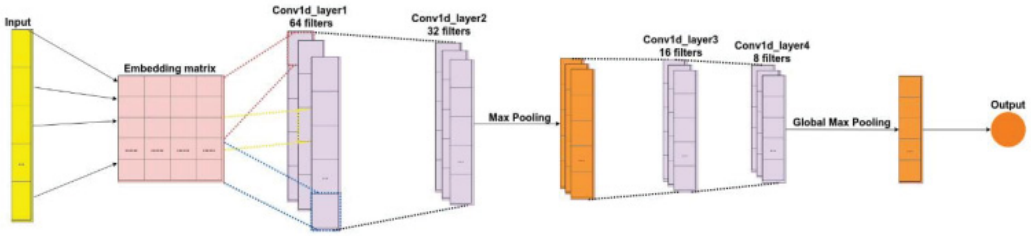

Figure 3. A convolutional neural network.

Figure 2.4: Convolutional Neural Network

2.3.3 Recurrent Neural Networks (RNN)

Recurrent Neural Network is an extension of feedforward neural network with an internal memory. [20] RNN is recurrent in nature since it performs the same function for each data input while the outcome of the current input is dependent on the previous calculation. After the output has been generated, it is duplicated and fed back into the recurrent network. For decision-making, it evaluates both the current input and the outcome from the prior input from which it has learnt. Long Short Term Memory Networks is an advanced RNN, a sequential network, that allows for the persistence of information. It is capable of resolving the gradient issue encountered by RNN. For permanent memory, a recurrent neural network, also known as RNN, is used.. As a result of the diminishing gradient, RNNs are incapable of remembering long-term dependencies. LSTMs are purposefully intended to prevent difficulties with long-term dependencies. Preprocessing the input data to reformat the data for the embedding matrix (the process is similar to the one described for the CNN). The next layer is the LSTM, which consists of 200 cells. The final layer is a completely interconnected layer with 128 text categorization cells. Given that there are two classes to be predicted, the last layer employs the sigmoid activation function [Fig 2.5] to decrease the vector of height 128 to an output vector of one (positive, negative).

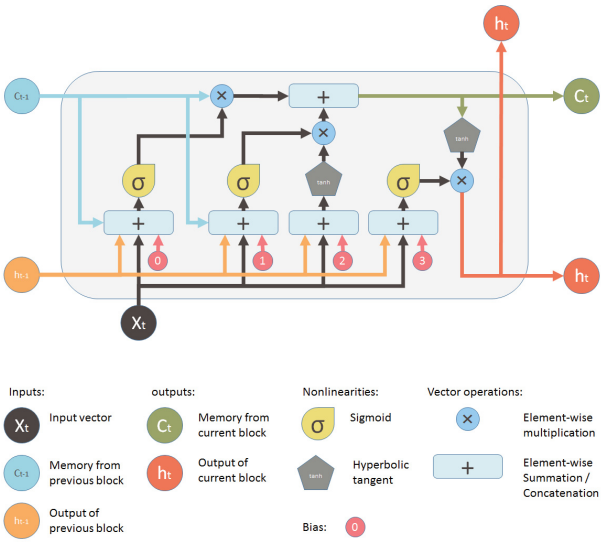

Figure 2.5: Recurrent Neural Network

2.4 Sentiment Analysis

Sentiment analysis is the technique of obtaining information about an entity and determining its subjectivities automatically. The purpose is to identify whether user-generated material expresses favorable, negative, or neutral sentiments. Classification of sentiment may be accomplished on three levels of extraction: aspect or feature level, phrase level, and document level. There are currently three solutions [24] to the issue of sentiment analysis : (1) lexicon-based strategies, (2) machine-learning-based techniques, and (3) hybrid approaches as shown in [Fig 2.6].

Initially, approaches based on a **lexicon** were utilized for sentiment analysis. They are separated into dictionary-based and corpus-based techniques [25]. In the first kind, sentiment categorization is accomplished by the use of a terminology dictionary, such as SentiWordNet and WordNet. However, corpus-based sentiment analysis does not rely on a predefined dictionary, but rather on a statistical analysis of the contents of a collection of documents, using techniques such as k-nearest neighbors (k-NN) , conditional random field (CRF) [22], and hidden Markov models (HMM) , among others.

Machine learning Techniques offered for sentiment analysis issues fall into two categories: (1) standard models and (2) deep learning models. Traditional models relate to traditional machine learning algorithms, such as the nave Bayes classifier , the maximum entropy classifier [21, 23], and support vector machines (SVM) . These algorithms receive as input lexical characteristics, sentiment lexicon-based features, parts of speech, as well as adjectives and adverbs. The precision of these systems relies on the selected characteristics. Deep learning models can deliver superior than traditional methods.CNN, DNN, and RNN are among the deep learning models that may be utilized for sentiment analysis. These methods handle categorization issues at the document, phrase, and aspect levels. The next section will cover these approaches of deep learning.

The hybrid techniques [26] combine methodologies based on lexicons and machine learning. Commonly, sentiment lexicons play a crucial part in the bulk of these tactics.

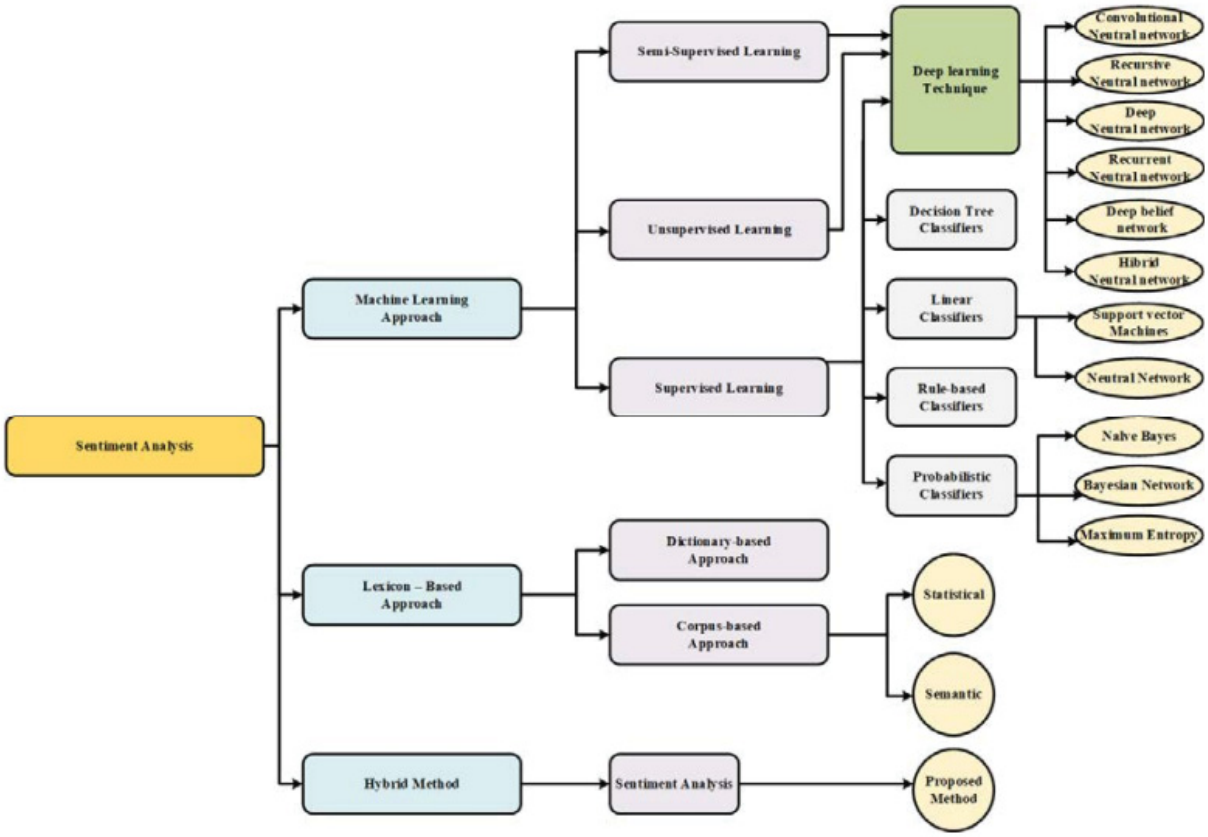

Figure 2.6: Categorisation of Sentiment Analysis Techniques

2.5 BERT

BERT is an open source natural language processing machine learning framework (NLP). Word embedding is intended to assist computers in understanding the meaning of ambiguous words in text by leveraging surrounding material to build context. [13]

BERT, which stands for Bidirectional Encoder Representations from Transformers, is based on Transformers, a deep learning model in which every output element is linked to every input element, and the weightings between them are produced dynamically depending on their relationship. (This is referred to as attention in NLP.)

A fundamental Transformer consists of an encoder that reads the text input and a decoder that generates a prediction for the job. Since the objective of BERT is to construct a language representation model, it simply requires the encoder. Encoder input for BERT is a series of tokens, which are transformed to vectors and then processed by the neural network. [14] Some of the other alternative options are Hugging Face:- Distilled BERT,GPT 23 and XLNet. They are efficient but BERT beat's them all and has been a better performer by being state of the art in 7 0f 11 NLP tasks.

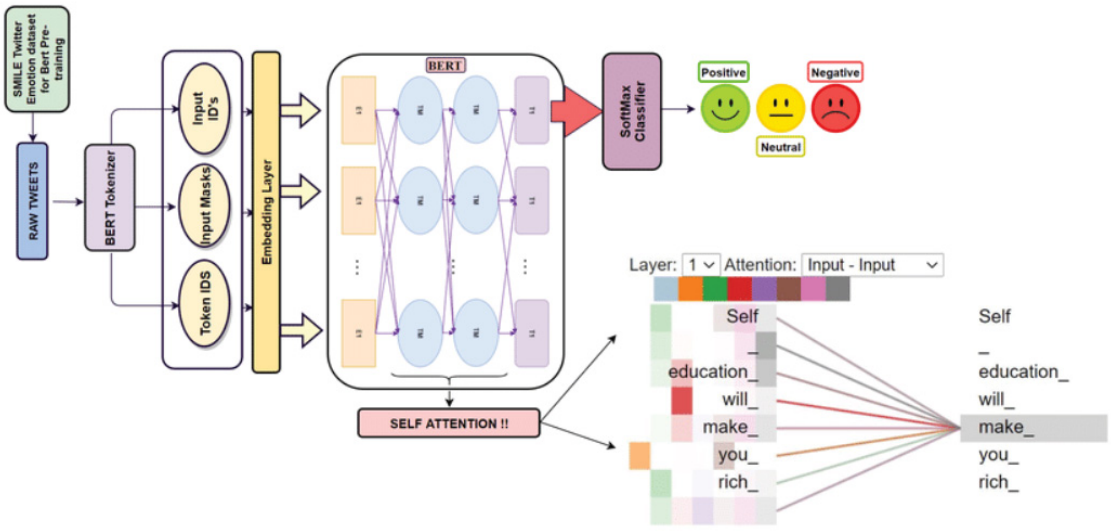

Figure 2.7: BERT Architecture

2.6 Performance Measures

The performance of the classifier is assessed using different metrics than the performance of the model. As the data used for the thesis consists of text data and

multi-class classification, the optimal metrics are selected. In multi-class classification, the accuracy, recall, F-score, and area under the curve are prominent measures (AUC). [17]

2.6.1 Accuracy

Classification accuracy is defined as the total number of correct predictions divided by the classifier's total number of predictions. A 'true positive' is a result when the model accurately predicts the positive class. Likewise, a 'true negative' is a result that the model accurately predicts the negative class. A 'false positive' result from the model mistakenly predicts the negative class as a positive class. A 'false negative 'result from the model mistakenly predicts the positive class as a negative class.For a binary classification problem the accuracy is calculated as in equation 3.32 [27].

Accuracy =

$$\frac{TP + TN}{TP + TN + FP + FN}$$

2.6.2 Precision

Precision is another statistic used to measure the performance of a classifier. It is determined by dividing the number of positive class predictions, i.e., true positives that belong to the positive class, by the total number of true positives.

Precision =

$$\frac{TP}{TP + FP}$$

2.6.3 Recall

Recall is calculated as true positive divided by the true positives and false negatives.

Recall =

$$\frac{TP}{TP + FN}$$

2.6.4 F1 Score

The F1 score is the harmonic mean of accuracy and recall, accounting for both measurements using the following equation: We use the harmonic mean rather than a simple average since it penalizes outliers. To design a classification model with the ideal combination of recall and accuracy, we maximize the F1 score.

F1 Score =

$$\frac{2 * Precision * Recall}{Precision + Recall}$$

Chapter 3

Related Work

This study aims to examine various techniques and methodologies in sentiment analysis that might serve as a reference for future empirical research.

Recently, deep learning models (such as DNN, CNN, and RNN) have been used to improve the efficiency of sentiment analysis jobs. In this part, cutting-edge techniques to sentiment analysis based on deep learning are examined.

Since 2015, several scholars have studied this tendency. Tang et al. [37]proposed deep learning-based algorithms for a variety of sentiment studies, including learning word embedding, sentiment categorization, and opinion extraction. Zhang and Zheng [26] addressed the use of machine learning to sentiment analysis. Both study teams employed POS as a text feature and TF-IDF to compute the weight of words for analysis. Sharef et al. [32] explored the advantages of large data sentiment analysis methodologies. The most recent studies [3, 7, 33] are cited in deep-learning-based techniques (namely CNN, RNN, and LSTM) were reviewed and compared with each other in the context of sentiment analysis problems.

Other research used sentiment analysis based on deep learning to many areas, including banking [2, 4] tweets about the weather [5], travel advisers, recommender systems for cloud services [34], and movie reviews [6, 29]. In [5], where text characteristics were automatically retrieved from several data sources, Word2vec was used to translate user information and weather knowledge into word embedding. Several papers [2, 34] use the same methodologies. Combining topic modeling with the findings of a sentiment analysis conducted on customer-generated social media data, Jeong et al. [43] highlighted product development prospects. It has been used as a tool for real-time monitoring and analysis of changing client demands in situations with fast-developing products. Pham et al. [35] analyzed travel evaluations and determined opinions for five criteria, including value, room, location, and cleanliness.

The application of polarity-based sentiment deep learning to tweets yielded [8, 9, 22]. The authors revealed how they employed deep learning models to boost the accuracy of their sentiment assessments. Most of the models are used for material posted in English, although there a handful that handle tweets in other languages, including Spanish , Thai , and Persian [36]. Researchers in the past have examined tweets using various models of polarity-based sentiment deep learning. Those models include DNN CNN , and hybrid techniques [9].

We discovered three prominent models for sentiment polarity analysis using deep learning from studies: DNN [15, 29], CNN [9], and hybrid [29]. In [7, 8, 37], CNN, RNN, and LSTM were evaluated independently on distinct data sets. However, a comparative comparison of these three methodologies was lacking. [8]

Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) exhibit very good overall accuracy [8, 29] when evaluating the performance of a single approach on a single dataset inside a certain domain (RNN). Hassan and Mahmood shown that CNN and RNN models may circumvent the short-text limitation of deep learning algorithms. Qian et al. [10] shown that Long Short-Term Memory (LSTM) performs well when applied to varying text levels of weather-and-mood tweets.

Recently, sentiment expression and categorization systems have acquired significant appeal. Numerous feature extraction methods are used in this study [21], including the Gabor filter, the Histogram of Oriented Gradient, the Local Binary Pattern, the Discrete Cosine Transform, and many more. The majority of approaches take the complete text as input, extracting characteristics and generating many sub spaces that are then used to assess different independent and dependent components [21].

The authors proposed deep learning sentiment analysis algorithms to classify Twitter data reviews [38]. Significant data demonstrate that deep learning outperforms conventional methods, such as Naive Bayes and SVM without Maximum Entropy. In their research, the authors have used LSTM and DCNN models. Using word2vec [21] to train word vectors for the DCNN and LSTM models. In this research, the Twitter dataset was used. This study shown that DNN is superior than LSTM for sentiment analysis using deep learning [38]. Furthermore, a big, almost meaningful data sample is required for mining.

Chapter 4

Method

This chapter covers the methodology used in the study. The research begins with a systematic assessment of the literature to identify frequently used algorithms for doing Sentiment Analysis. It is followed by experiments to compare algorithm performance. Experiments, Case studies, and Surveys [44] are the most prevalent empirical methods.

This research may be conducted objectively by any industry or applied to many topics, and the results can be used appropriately. Consequently, experimentation is selected as one of the study methods.The experimentation method [44] is an analytical and scholarly strategy in which the researcher systematically conducts an experiment The primary purpose of experimentation is to apply and assess the chosen algorithms using defined evaluation procedures. In addition, our dataset includes dependent and independent variables, which motivates us to do experiments.The experimental research approach is used to answer research questions 2 and 3, The Experiments utilize the same hardware and software described in this chapter.

4.1 Literature Review

A Systematic Literature Review (SLR) is a way of examining the existing literature on a given research issue or subject that is systematic, rigorous, and transparent. It entails doing systematic and organized searches for, selection of, appraisal of, and synthesis of relevant research. An SLR's goal is to offer an overview of existing knowledge on a field, identify gaps in the literature, and influence future research objectives. It is most often utilized in many fields, although it may be employed in any sector.

For RQ1, literature research helps us uncover deep learning approaches and choose the most popular algorithms. We examine relevant papers to determine which metrics to utilize to evaluate the algorithm for efficient sentiment analysis. Research will build on the literature study findings.

How to do SLR:

- Choose keywords that relate to the thesis.
- Make a short list of the helpful resources that have something to do with the thesis.

- Add the criteria for which articles are included and which are not.
- Choose the research papers and publications that will help with the thesis.
- Look at the papers you found after your search.
- Write a summary of what you found and use it in the next steps of your research

SpringerLink,IEEE Xplore,Google Scholar, ACM Digital Library, arXiv, IGI Global, and others are searched using a search string. Snowballing, which finds relevant new articles by referencing previously recognized documents, also finds papers. RQ1 literature search strings are below.

Search terms and phrases:

- 'Sentiment Analysis','Opinion Mining','Sentiment Classification','Machine Learning','Deep Learning','Supervised Learning','Word Embedding','Neural Networks'
- (Sentiment Analysis OR Opinion Mining OR Opinion Analysis OR Sentiment Classification) AND (Machine Learning OR Deep Learning OR Supervised Learning) AND (Word Embedding) AND (Challenges) AND (Opportunities)

The inclusion criteria for the articles are:

- The articles must be written in English as their primary language.
- Articles considered for inclusion are required to make sentiment analysis the primary focus of their discussion.
- Articles published in the last 10 years.
- Articles are required to address at least one of the research questions.
- Article must be published in top-tier conferences, publicationsand journals.

The exclusion criteria for the articles are:

- Exclude non-English-written articles.
- Additionally omitted are paid papers, student articles, and older versions which have various versions.
- Articles published before 2013 are excluded.
- Articles unrelated to the present research.

SLR is used to answer the RQ1

- **RQ1:** Which are the popular deep learning techniques used to perform sentiment analysis?

4.2 Experiment

Hardware Environment

Table 4.1: Hardware Environment.

Processor	Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz 2.5
OS	Windows 10
Memory	64gb

4.2.1 Software Tools

Python is selected as the experiment's programming language since it supports a broad variety of libraries required to implement sentiment analysis, extract data, etc. Python [40] is useful for site scraping, data preprocessing, prediction, and visualization studies. Python is an open-source programming language with excellent frameworks for Artificial Intelligence, Machine Learning, statistical analysis, and visualization. It supports a variety of libraries with robust capabilities and highly customized implementations; some packages are used for improved outcomes

4.2.1.1 Pandas

For data pre-processing and data handling, panda's package is used. [40] For the next steps, it is very important to create a data structure for the scrapped data that is provided by pandas for fast and flexible structuring of data. Its multipurpose functionality for handling data is an advantage, all the data that is scraped for the thesis work is converted to a data frame for further analysis and prediction [40].

4.2.1.2 NumPy

A NumPy stands for "Numerical Python" is used for implementing numerical computations for vectors and matrices. It provides 50 times faster computation than list data. For data analysis and numerical calculation in the thesis, this library is used [38].

4.2.1.3 nltk

Natural Language Toolkit (NLTK) is a standard library that eases the use and implementation of natural language processing and information retrieval tasks like tokenization, stemming, parsing, and semantic text relationships [40].

4.2.1.4 Sklearn

Scikit-learn provides tools and functionality for machine learning and statistical modeling for classification, clustering, and other predictions. For example, split data into train, validation, and test subsets, create features for text inputs, create tokens, and count vectors like frequency count for tf-idf. For classification, task data is split into train and test [41].

4.2.1.5 seaborn

Like matplotlib, the seaborn library is also used for data visualization and exploratory data analysis, built on Matplotlib to create customized plots [40].

4.2.1.6 TensorFlow

TensorFlow is an end-to-end open-source library for creating deep learning models to handle extensive data and implementing complex models like BERT to simplify and speed up the process [18].

4.2.1.7 Keras

Like tensorflow, Keras is an open-source software high-level Application Programming Interface (API) that provides a Python interface for artificial neural networks., it acts as an interface for the TensorFlow library. It is more user-friendly and a little faster compared to Tensor flow. For the implementation of the Bidirectional Encoder Representations from Transformers (BERT) model in the thesis, this package is used [18]

4.2.2 Data

In order to conduct sentiment analysis, researchers may either create their own data or make use of already databases Creating a new dataset allows for the use of data that is relevant to the issue being analyzed, and the usage of personal data guarantees that no privacy rules are broken [29].

The purpose of this thesis was to collect data and public opinion on the furniture shop through various social media outlets. As part of the process, we have collected data and built datasets from Twitter, Reddit, and several consumer forum websites that include reviews of the furniture store's items. Web scraping was utilized to produce the dataset.

4.2.2.1 Web Scrapping

Web scraping is the automatic collection of web data and information. It is essentially the extraction of web data. Website Scraping is concerned with information retrieval, newsgathering, web monitoring, and more. [15] Utilizing web scraping allows accessing the large quantity of information available online quick and straightforward. It is far quicker and less complicated than manually pulling data from websites.

There are mainly two ways to extract data from a website:

- Use the API of the website (if it exists). For example, Twitter has the Twitter API which allows retrieval of data posted on Twitter
- Access the HTML of the webpage and extract useful information/data from it. This technique is called web scraping or web harvesting or web data extraction.

4.2.2.2 BeautifulSoup

Beautiful Soup offers straightforward techniques for exploring, finding, and editing a parse tree in HTML and XML files. It converts a complicated HTML page to a Python object tree. It also transforms the page to Unicode automatically, so you don't have to worry about encodings. This program allows you not only scrape data but also clean it. Figure 4.1 shows an example of the implementation of this library. [16]

```
r = requests.get('https://www.reviews.io/company-reviews/store/ikea')
soup = BeautifulSoup(r.text, 'html.parser')
regex = re.compile('.*Review__body u-wordBreak--wordBreak.*')
results = soup.find_all('span', {'class':regex})
reviews = [result.text for result in results]
```

Figure 4.1: Web Scraping of the Website 'www.reviews.io' done using beautifulsoup library

Some of the other method's that I have used for the data collection are tools like **OctaParse**, is a cloud-based online data extraction system that enables users to collect pertinent data from a variety of websites. It allows users from many sectors to scrape unstructured data and store it in a number of forms. I have also considered using **Tweepy**,an open-source Python program that facilitates easy access to the Twitter API using Python which is useful in extracting tweets. During web scraping, a lot of different non-necessary attributes, like user id and time of post, are pulled out. So, we got rid of those columns and cleaned up our table so that the main attribute is the review of the furniture store.

	review
0	"My partner and I visited Lakeside IKEA and wa...
1	"What a horrible experience after collecting I...
2	"Same as many others have found, disgraceful c...
3	"HONESTLY I WISH I COULD WRITE -5 STARS BECAUS...
4	"Well! I have approached Ikea a few times and ...

Figure 4.2: This is the sample Dataset that is formed after the web scraping one of the social media channel.

4.2.3 Dataset

The datasets were gathered from various sites and are about different things regarding the topic so that a wide range of experiments can be done. Because of this, the results have made it possible to compare the performance of deep learning models in sentiment analysis in a wide range of ways. The following explains these data sets:

- **Twitter** Dataset, is the primary dataset. It included close to 1.2 million tweets that discussed various opinions and thoughts about the furniture store. It had various fields like the 'user id', 'date', 'tweet url','text', which contained the main review.

- **Reddit** Dataset, has been obtained from the social networking site Reddit, using the search string of the furniture store. This dataset has around close to 2200 samples.

- **Sitejabber** Dataset, has comments from customers regarding the product. This dataset has 9890 samples. It has the same fields as the Twitter Dataset.

- **Reviewsio** Dataset has a collection of reviews about the store. We have about 23100 samples.

- **Consumer Affairs** is the dataset that has the most extensive history of reviews. There are around 51000 samples in all.

A representative sample of tweets taken from one datasets is shown in Figure 4.3. It includes data pertaining to all of the following areas:

- "review author name link" is the Handle Name of the User.
- 'review title link href' is the Unique link to that Comment.
- 'ReviewTitle' is the title of the review.
- 'review date' is the date of the review.
- 'review text' is the text of the review.

In order to carry out the experiment, we made use of the "text" field from the Dataset's.

review__author__name__link	review__title__link href	ReviewTitle	review__date	review__text
LincolnÂ A.	https://www.sitejabber.com	Good quality	March 7th, 2021	My purchase was a reading desk. They
RichÂ B.	https://www.sitejabber.com	I like its minimalism in	August 2nd, 2020	I bought some furniture for the childre
ClaireÂ B.	https://www.sitejabber.com	Don't give Ikea your ca	May 11th, 2021	Not only have Ikea failed to deliver my
SophiaÂ M.	https://www.sitejabber.com	This brand is very popu	November 28th, 2020	This brand is very popular now. I really
AJÂ W.	https://www.sitejabber.com	You get what you pay	June 9th, 2020	Some good, some bad. Honestly it's a
NancyÂ N.	https://www.sitejabber.com	Horrible service and pr	October 14th, 2020	We ordered a shelf from IKEA in vaugh
KimÂ B.	https://www.sitejabber.com	IKEA is good, but the ca	August 26th, 2021	This review is more about the compan
AlberteÂ O.	https://www.sitejabber.com	Excellent	June 1st, 2020	Nothing to say except the words of gra
LaurieÂ B.	https://www.sitejabber.com	Beautiful displays but t	April 18th, 2019	I visited the Oak Creek store in Wiscon
maeÂ q.	https://www.sitejabber.com	Ikea delivery	November 2nd, 2021	My first time using delivery service fro
GioiaÂ L.	https://www.sitejabber.com	Amazing shop	September 14th, 2021	This is am amazing shop. There are so
AnnaÂ D.	https://www.sitejabber.com	Doesn't get any worse	July 15th, 2020	I've been trying to get in touch with

Figure 4.3: Sample of Sitejabber Dataset

4.2.4 Data Preparation

The scope of our experiment requires the data to be labelled in the data set and classify it. As discussed in the thesis above we have used the BERT transformer to label the data into positive and negative based on the polarity generated for the tweet.

- Download and extract the dataset, then explore the directory structure.
- Here you can choose which BERT model you will load from TensorFlow Hub and fine-tune. There are multiple BERT models available.
- We have used the **bert-base-multilingual-uncased-sentiment** that has helped us in the classification as shown in figure 4.3

Using this BERT Model we have been able to classify the reviews into 1-5 as shown in figure 4.4. This has been translated into positive and negative.

```
In [5]: tokenizer = AutoTokenizer.from_pretrained('nlptown/bert-base-multilingual-uncased-sentiment')

        model = AutoModelForSequenceClassification.from_pretrained('nlptown/bert-base-multilingual-uncased-sentiment')
```

Figure 4.4: BERT Model

	review	sentiment
0	"My partner and I visited Lakeside IKEA and wa...	2
1	"What a horrible experience after collecting I...	1
2	"Same as many others have found, disgraceful c...	1
3	"HONESTLY I WISH I COULD WRITE -5 STARS BECAUS...	1
4	"Well! I have approached Ikea a few times and ...	1
5	"I don't know where to begin from! Worse worse...	1
6	"I would gone 0 stars, but I have to give 1 st...	1
7	"Wow, where to start with my utter disdain and...	1
8	"I have been waiting since before Christmas fo...	2
9	"I'm an old Ikea customer going back to 1994, ...	1
10	"HORRIBLE EXPERIENCE! Do not order your kitche...	1
11	"Ikea Exeter, please see today's offer for Pax...	5
12	"Delivery costs are ridiculous, stuff you woul...	1
13	"I'm sooooooooo extremely disappointed with I...	1
14	"I am writing here to comment on your company'...	1
15	"So they let you place an order - then turn ar...	1

Figure 4.5: A sample of the labelled dataset generated

4.2.5 Training, Validation, and Test sets

The datasets are randomized split into three subsets: training, validation, and testing, with a percentage of 70:10:20, respectively. All of the subsets must have the same number of positive and negative comments, so the percentage of each class is kept by using the scikit-learn function train_test_split() with the parameter stratify The model is given the training subset. The validation subset is used to figure out when to stop early, and the testing sub - set is used to check how well a model works. In Table 4.2, you can see how big each subset is.

Dependent variables: The Performance Metrics used. **Independent variables:** Deep Learning Algorithms and Word Embedding Techniques used.

Table 4.2: Twitter Data Set: Sizes of each subset

Subset	Number of Tweets
Training Set	861000
Validation Set	123000
Testing Set Set	246000

4.2.6 Data Cleaning

Text cleaning is a preprocessing step that removes words or other components that do not contain relevant information, and thus may reduce the effectiveness of sentiment analysis. Text or sentence data include white space, punctuation, and stop words. Text cleaning has several steps for sentence normalization. [9] All datasets were cleaned using the following steps:

4.2.6.1 Tokenization:

Separating the phrase into words.

4.2.6.2 Lower casing:

Lowercase conversion of a word (THESIS -> thesis).Name and name have the same meaning, but when not changed to lower case, they are treated as separate words in the vector space model (resulting in more dimensions).

4.2.6.3 StopWords

Stop words are widely used terms (a, an, etc.) that are eliminated from papers. These words have no practical significance since they do not discriminate between two papers.

4.2.6.4 Stemming:

It is the transformation of a word into its basic form.

4.2.6.5 Lemmatization:

Lemmatization, unlike stemming, reduces words to an existing term in the language. The construction of a stemmer is simpler than that of a lemmatizer, since the latter needs extensive understanding of linguistics for developing dictionaries that search up the lemma of a word.

4.2.7 Word Embedding

Most machine learning algorithms cannot interpret strings or plain text in its basic form. In contrast, they need numerical inputs to work. By translating words into vectors, word embeddings enable the processing of massive amounts of text data and their adaptation to machine learning algorithms.Encoded so that 1 represents the spot where the word exists and 0 represents all other positions. Our thesis lays out a major focus on the usage of DNN, CNN, and RNN algorithms with word embedding and TF-IDF -IDF. **Word2vec**, a popular word embedding technique,is used here alongside TF-IDF.The whole text is examined, and the vector construction procedure is carried out by identifying the words with which the target word appears most frequently [3]. In this manner, the semantic proximity of the words is also shown, unlike other techniques A procedure of unsupervised learning is carried out. Using artificial neural networks, unlabeled data is used to train the Word2Vec model that creates word vectors.

4.2.8 TF-IDF

TF-IDF measures the mathematical importance of document words[2]. Vectorization resembles OHE. Instead of 1, the word's value is TF-IDF. Multiplying TF and IDF yields.Term frequency is the ratio of target terms to overall terms in the document.IDF is the logarithm of the ratio of total documents to target-term documents.We utilized the vectorizer class from the scikit-learn package for TF-IDF. The formula that is used to compute the tf-idf for a term t of a document d in a document set is tf-idf(t, d) = tf(t, d) * idf(t), and the idf is computed as idf(t) = log [n / df(t)] + 1 , where n is the total number of documents in the document set and df(t) is the document frequency of t;

4.2.9 Implementation

Depending on the dataset, a particular processing approach was then used to ease model construction Using instance, Twitter dataset, we eliminated columns that are not relevant for sentiment analysis: "id", "date", "query stalongsidend "username" and transformed class label to positive and negative values.

After cleansing the datasets, sentences were separated into individual words and returned. to their fundamental form by lemmatization. At this step, phrases were transformed into continuous vectors Using two approaches, word embedding and TF-IDF, we can convert feature vectors into vectors of words. Inputs to the deep learning algorithms assessed in this research were both types of feature vectors. These were the Convolutional Neural Networks, Deep Neural Networks, and Recurrent Neural Networks algorithms. Thus, construction using models were generated, one for any kind of vector.

The majority of conventional models use well-known characteristics, like bag-of-words, n-grams, and TF-IDF. Such characteristics disregard semantic similarity between words.Currently, several deep learning models in natural language processing need word embedding findings as input characteristics. Since neural networks may be used to vectors sentiment via word embedding, we utilize Word2vec for training initial vectors using the datasets given above.

Like mentioned earlier k fold cross validation with k equal to ten is used to determine the efficacy of various embeddings. Initialized with random weights, the function layer, the embedding layer that understands the embedding for all terms in the training datasets. In this instance, the vocabulary size is 17000, the highest len is 40 characters. The output is a 40 by 300 matrix.

Initial 1D CNN layer contains a filter with a size of 3 kernels. For this, 64 filters will be defined. This permits 64 distinct features to be trained on the initial layer Consequently, the output of the first neural network layer is a 40 64 neuron matrix, and the output of primary CNN fed into the next one. ˜
˜

Again, 32 distinct filters will be defined for training on this level. Using the same reasoning like the primary layer, resulting matrix will have dimensions of 40 by 32.M The max pool layer is often employed after a CNN layer to minimize the output's complexity and avoid data overfitting. In this instance, we select a level of three. This indicates that the output matrix size of this layer is 13 by 32. 13×16 matrix and a 13×8 matrix come out of the third and fourth 1D Convolutional Neural Network layer. ˜
˜

Avg pooling layer that is used to prevent overfitting. We w,ill utilize the average value rather than the highest number in this instance since it will provide superior results. The size of the output matrix is 1 by 8 neurons. a fully connected layer which has sigmoid activation is the last layer that reduces the 8-dimensional vector to 1 for prediction ("positive," "negative").

Chapter 5

Results and Analysis

In this chapter, we show the results of the Literature Review, the Experiments, and the analysis performed in response to the research questions.

5.1 Systematic Literature Review Results

Table 5.1: SLR Results

No	Article	Results
1	Twitter sentiment analysis with a deep neural network: An enhanced approach using user behavioral information [42]	CNN was the approach that was used for the study work that was done on Twitter sentiment analysis. The dataset that was utilized was from the SemEval 2016 workshop, and the goal of the experiment was to extract features based on information on user behavior.
2	Sentiment analysis through recurrent variants latterly on convolutional neural network of Twitter. [9]	Sentiment analysis using recent recurrent variations was the focus of his work. CNN and RNN are the techniques that are being employed in an effort to create domain-specific word embedding on Twitter.
3	Big Data: Deep Learning for financial sentiment analysis [2]	Deep learning was the topic of this paper's emphasis for the analysis of financial sentiment. LSTM, Word2vec, and CNN were among of the techniques that were used. These were applied to the dataset of StockTwits with the goal of enhancing the effectiveness of the sentiment analysis for StockTwits.
4	Sentiment Analysis of a document using deep learning approach and decision trees [45]	Google's Word2Vec-aided deep learning sentiment analysis. Preprocessing extracts characteristics initially. Word2Vec uses CBOWs to forecast the current word and skip-grams to anticipate the surrounding words. Data is trained using an Elman-type RNN and clustered. Deep learning outperformed CBOW, although the accuracy difference was small.
		Continued in next page

Table 5.1 Continued

No	Article	Results
5	Deep Learning in Sentiment Analysis. In Deep Learning in Natural Language Processing. [37]	The author has addressed the topic of attempting to improve the current state of the art in a variety of deep learning sentiment analysis tasks. CNN, DNN, and RNN are examples of popular deep learning algorithms utilized today. The dataset was compiled using information acquired from a wide variety of social networking sites. Discussions have taken place on the classification of sentiments, the extraction of opinions, and the fine-grained analysis of sentiments.
6	Enhancing deep learning sentiment analysis with ensemble techniques in social applications. [38]	The goal of the research was to improve the effectiveness of deep learning sentiment analysis in social applications. In this study, a deep-learning-based sentiment classifier using a word embedding model and a linear machine learning method was applied. This research used the datasets Google Reviews , Vader, IMDB, Stanford Movie Reviews, and Sentiment140. This study aimed to enhance the performance of deep learning methods and merge them with conformal approaches based on manually derived features.
7	Social media mining for product planning: A product opportunity mining approach based on topic modeling and sentiment analysis. [43]	A method for mining product opportunities based on topic modeling and text analytics is the objective of this study. Utilized techniques include topic modeling based on LDA, sentiment analysis, and the opportunity algorithm. The information source is social networking sites Twitter, Facebook, Instagram, and Reddit. Aiming to identify product development possibilities from social media data supplied by customers
8	Application of deep learning to sentiment analysis for recommender system on cloud [12]	Analysis of user sentiment performed by a recommender system hosted in the cloud RNN and naive Bayes classifier are the two algorithms that are used. Amazon dataset has been utilized. The purpose of this paper is to recommend locations that are close to where the user is currently located by doing an analysis of the various reviews and then generating a score based on the information gleaned from those reviews.
		Continued in next page

Table 5.1 Continued

No	Article	Results
9	A recursive deep learning model for opinion mining in Arabic as a low resource language [46]	The author's motivation for writing this article is to do opinion mining , which is a low-resource language. The strategy that will be used is known as recursive deep learning. The dataset was comprised of online commentary taken from QALB, Twitter, and MSA-formatted articles published by Newswire. He intended to do the following: Supplying the autoencoder with input characteristics that were more extensive and exhaustive, and carrying out semantic composition.
10	Sentiment analysis in for improvement of products and services: A deep learning approach. [36]	Analysis of customer sentiment for the sake of product and service improvement is the author's motivation.The approach that will be used is CNN combined with Word2vec. The dataset consists all of tweets sent in Spanish on Twitter. His objective was to determine the level of pleasure felt by customers and to identify areas in which goods and services may be enhanced.
11	Neural Networks and Deep Learning [19]	This article presents the results of a survey about deep learning for On social networking sites, we do CNN, DNN, RNN, and LSTM analysis of sentiment in addition to other similar approaches. The sentiment analysis using words will be the focus of the study. analysis of sarcasm, embedding, etc. a multimodal approach to the study of feelings information necessary for analyzing attitudes and feelings
12	Review Sentiment Analysis Based on Deep Learning [47]	In this, an unsupervised Hierarchical Deep neural network is developed for document-level sentiment analysis. The findings are compared with those of a support vector machine (SVM), and it is determined that the neural network delivers better accurate results as the size of the dataset expands.
13	Deep Learning for sentiment Analysis On Google Play Consumer Review [48]	Chinese Google Play reviews are analyzed for sentiment. The experiment included 196,651 web crawler reviews. Pre-processing and dictionary integration followed. Comparing LSTM, Naive Bayes, and SVM classifiers. LSTM's 94% accuracy outperformed SVM's 76.46% and Naive Bayes' 74.12%.

5.1.1 SLR Analysis

The list of articles collected from the SLR can be found in the table that can be seen above. This SLR was helpful in determining the most suitable method for this study and in providing a solution to RQ1. The authors of these studies made extensive use of deep learning methods while doing sentiment analysis. After reviewing the relevant literature, we determined that DNN, CNN, and RNN are the models that see the greatest application in the field of sentiment polarity analysis. On the other hand, there is not enough research done to provide a comprehensive comparison of these common approaches.

5.2 Experiment 1 - Results

Experiments with the various datasets outlined above were carried out using DNN, CNN, and RNN models in order to evaluate the effectiveness of those methods while using TF-IDF feature extraction as well as word embedding. The results of these experiments are shown in the next section. In each experiment, the code's parameters are set. Area under Curve, F-score ,Accuracy were utilized across all experiments to assess the performance of the models.Twitter dataset is the first dataset that was analyzed. Its content was designated as either positive or negative. Since this dataset comprises a much higher number of tweets than the previous datasets, we investigated the performance of models produced from subsets containing varying proportions of the starting data. The images below are a set of performance metrics for the amount of tweets(Percentage). The x-axis represents the percent of the dataset that has been processed(From 0 to 100%) and the y-axis is the metric values from 0 to 1.

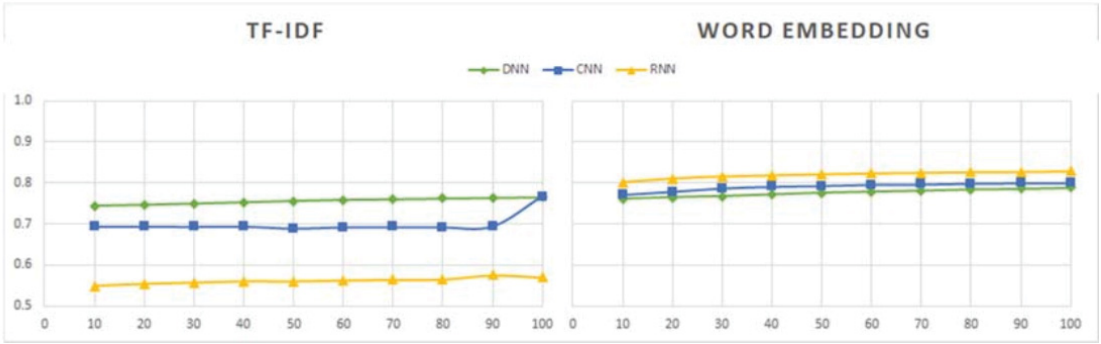

Figure 5.1: Accuracy values of the models with TF-IDF and WordEmbedding

The first performance metric, Accuracy, has been made, and Figure 5.1 shows how the varying parts of the data all follow the same pattern for both TF-IDF Word Embedding, with the exception of CNN for TF-IDF, which goes up exponentially as it gets closer to the last 10%. CNN's accuracy with TF-IDF was 0.7 most of the time and 0.8 at the end. With Word Embedding, CNN has always been right around 0.8 of the time. Both TF-IDF and Word Embedding gave DNN a score of between 0.75 and 0.8. With an average accuracy of around 0.55, RNN with TF-IDF did the worst. With an accuracy of over 0.8, Word Embedding with RNN has done better than everyone else.

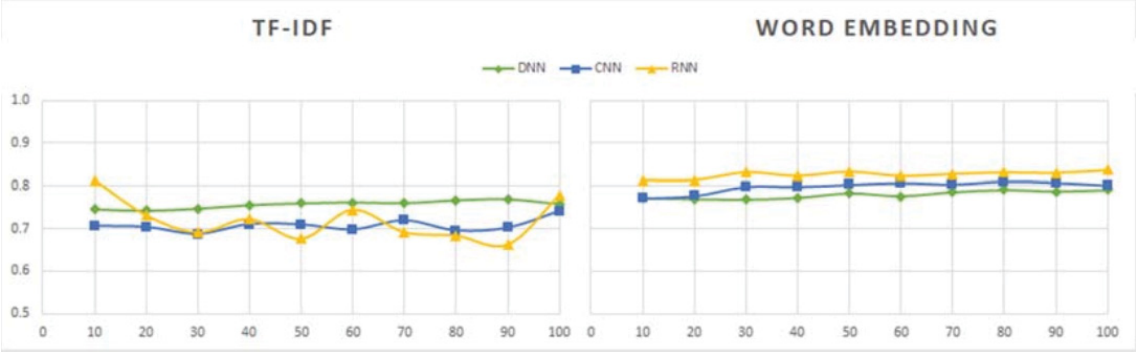

Figure 5.2: Recall values of the models with TF-IDF and WordEmbedding

Both TF-IDF and Word Embedding gave DNN a Recall Value between 0.75 and 0.8. CNN performed slightly better with Word Embedding with Recall Value closer to 0.8 in comparison to TF-IDF which was close to 0.7.RNN with Word Embedding was again the best performer with a value above 0.8 however it had a really nonlinear trend with TF-IDF as seen in Figure 5.2

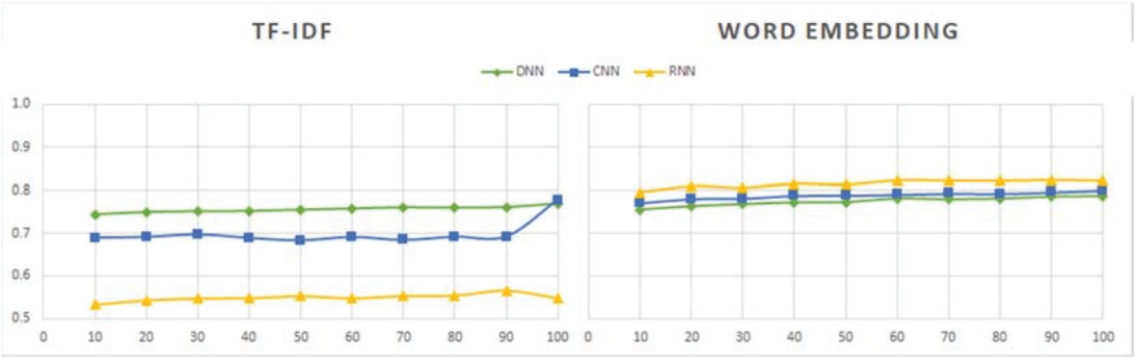

Figure 5.3: Precision values of the models with TF-IDF and WordEmbedding

Figure 5.3 shows that Word Embedding has consistently performed with CNN, RNN, and DNN with an approximate precision of 0.8. TF-IDF was able to equal this value with DNN, but it failed badly with RNN, scoring a dismal 0.5. CNN performed decently, with precision ranging from 0.7 to 0.8.

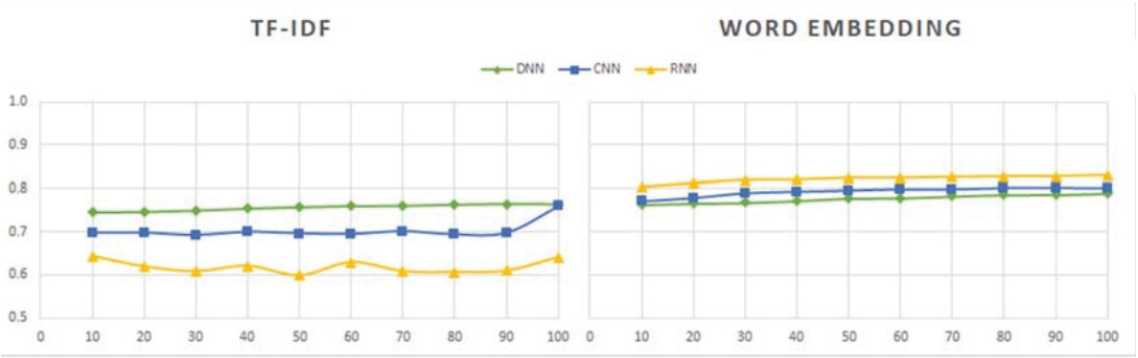

Figure 5.4: F-Score values of the models with TF-IDF and WordEmbedding

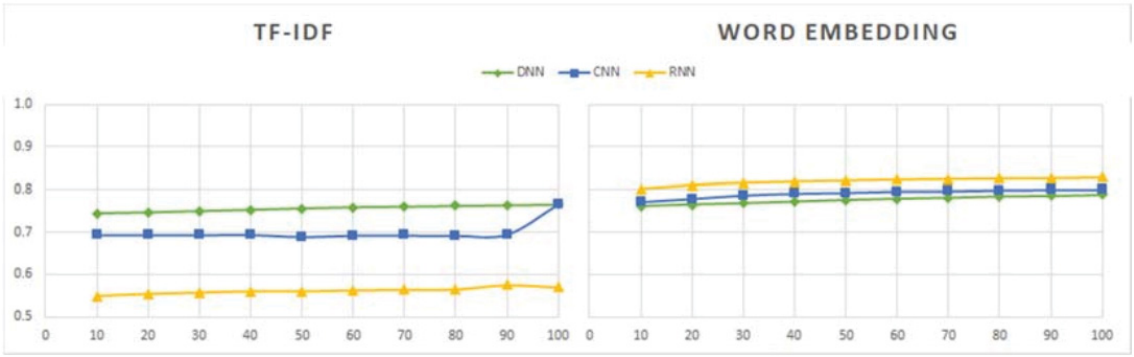

Figure 5.5: AUC values of the models with TF-IDF and WordEmbedding

Figures 5.4 and 5.5 illustrate a similar pattern for F-Score and AUC. The average value for Word Embedding is around 0.8, with RNN doing the best, followed by CNN and DNN. DNN has the highest TF-IDF score of 0.75, followed closely by CNN with 0.7. RNN has performed badly once again, with an AUC value of 0.55 and an F-Score of 0.6.

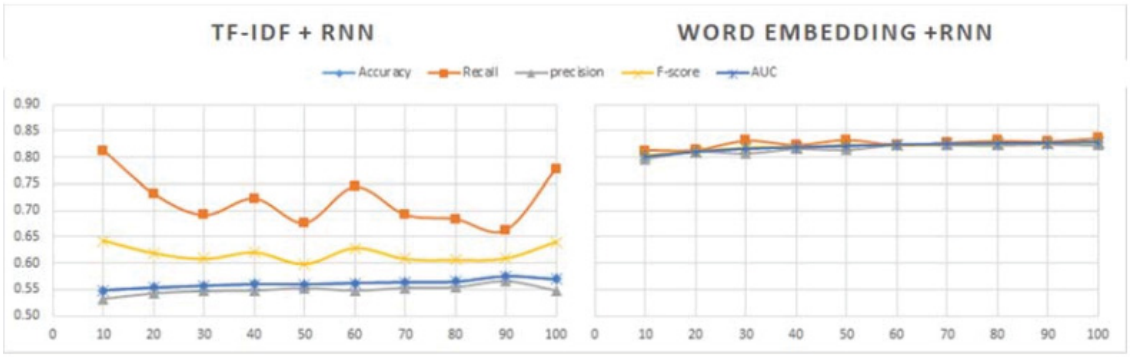

Figure 5.6: Performance measures of the Recurrent Neural Network with each of the Word Embedding Technique

Figures 5.1–5.6 clearly demonstrate the superior efficiency of the models while utilizing word embedding versus TF-IDF for all studied criteria. This increase is particularly noteworthy for Recurrent Neural Network, which is the algorithm that produces the greatest results when combined with word embedding. In contrast, RNN is the least effective of the three algorithms examined when combined with TF-IDF. Figure 5.6 illustrates the metric values produced by RNN models.

In the graphs above, we can also see that when it comes to word embedding, there aren't any big differences between the values of the evaluation measures for the three deep learning methods. However, when it comes to TF-IDF, the differences between the three methods are noticebale.Regarding the size of the dataset, its effect on the outcomes is negligible for word embedding but somewhat stronger and inconsistent for the TF-IDF approach.

Based on the study of the Twitter dataset's findings, we can conclude the following, word embedding is a better approach than TF-IDF. Furthermore, its use would provide us to deal with a small set of data representing fifty percent of the overall sample at a reduced computing cost and with negligible differences in the outcomes.

The tables below show the result's of the dataset's used.

Table 5.2: Twitter Dataset

Metrics	TF-IDF			word2vec		
	CNN	DNN	RNN	CNN	DNN	RNN
Accuracy	0.7563	0.7548	0.5432	0.8001	0.7702	0.815
Recall	0.7321	0.7423	0.7623	0.8012	0.7865	0.8241
Precision	0.7366	0.748	0.7635	0.8023	0.7845	0.8269
F Score	0.7542	0.764	0.6412	0.8074	0.7888	0.818
AUC	0.754	0.746	0.7557	0.8006	0.7875	0.8214

Table 5.3: SiteJabber Dataset

Metrics	TF-IDF			word2vec		
	CNN	DNN	RNN	CNN	DNN	RNN
Accuracy	0.6651	0.6924	0.5421	0.7142	‘0.7012	0.7567
Recall	0.6678	0.6932	0.8421	0.7241	0.7004	0.8014
Precision	0.6623	0.7012	0.4327	0.7123	0.7023	0.7423
F Score	0.678	0.6978	0.5874	0.7145	0.7088	0.7784
AUC	0.6642	0.6933	0.5023	0.7414	0.7023	0.762

Table 5.4: Reddit Dataset

Metrics	TF-IDF			word2vec		
	CNN	DNN	RNN	CNN	DNN	RNN
Accuracy	0.7124	0.7542	0.5062	0.7541	0.7325	0.7247
Recall	0.7244	0.7321	0.6231	0.8102	0.7294	0.7384
Precision	0.7144	0.7655	0.5541	0.7321	0.7325	0.7221
F Score	0.7144	0.7452	0.5210	0.7622	0.7322	0.7215
AUC	0.7211	0.7451	0.501	0.7514	0.7358	0.7214

Table 5.5: Review IO Dataset

Metrics	TF-IDF			word2vec		
	CNN	DNN	RNN	CNN	DNN	RNN
Accuracy	0.8122	0.8412	0.5594	0.8547	0.835	0.864
Recall	0.7954	0.8321	0.4512	0.8324	0.8365	0.8547
Precision	0.8255	0.8411	0.6021	0.8542	0.8369	0.8632
F Score	0.8011	0.8423	0.4523	0.8471	0.8214	0.874
AUC	08114	0.8544	0.5513	0.8541	0.8331	0.8641

Table 5.6: Consumer Affairs Dataset

Metrics	TF-IDF			word2vec		
	CNN	DNN	RNN	CNN	DNN	RNN
Accuracy	0.7921	0.8423	0.5741	0.8142	0.8014	0.8563
Recall	0.7412	0.8654	0.5541	0.8214	0.7714	0.8741
Precision	0.8241	0.8365	0.5847	0.8102	0.8001	0.8475
F Score	0.7845	0.8425	0.5632	0.8102	0.7821	0.8541
AUC	0.795	0.8475	0.5741	0.8147	0.7956	0.8547

The findings drawn from the study of Twitter dataset are simply validated by the results of the additional datasets. In general, the pairing of RNN with word embedding exhibits the best performance, however there are outliers. These are generated in "Reddit", in which the values of all metrics, barring recall, are slightly greater for DNN with TF-IDF than it is for RNN with word2vec. CNN and Word2vec provided the top results for Recall, Precision, and AUC for Reddit. These are the minuscule distinctions between the biggest and smallest datasets. Similarly, as stated before, we can confirm that word embedding is a more suitable approach than TF-IDF for doing sentiment analysis, inspite of the tiny gains observed with TF-IDF for particular data sets.

5.3 Experiment 2 - Results

After examining the findings regarding the accuracy of the forecasts, it is required to collect information on the computing cost connected with the creation of models, since the discrepancies between the results, or between some of the results, aren't particularly noteworthy. The objective is to determine if the best reliable values are attained at a more or less computational cost. Table below illustrate the CPU timings. The processing time needed to generate models from the datasets.

The tables demonstrate that TF-IDF, which generates less accurate models, consumes more computing time than word embedding. This is another reason why this last strategy is the most recommended. With both TF-IDF and word embeddings, RNN is more time-taking approach. Given that the advances of RNN relative with CNN, DNN are not particularly substantial in the latter situation, the usage of DNN and CNN might be seen more acceptable when reducing computational cost is a priority.

Dataset	TF-IDF			Word Embedding		
	DNN	CNN	RNN	DNN	CNN	RNN
Twitter Dataset	*10 min 23 sec*	*34 min 16 sec*	***1 hour 32 min 14 sec***	*4 min 08 sec*	*6min 12 sec*	***1hour 32 sec***
Sitejabber Dataset	*20.8 s*	*14.2s*	*29 min 5 s*	*11 s*	*17.26 s*	*3min 10s*
Reddit Dataset	*10.2 s*	*9.52s*	*19min 3s*	*14.1s*	*19.4 s*	*1min 42 s*
Review.io Dataset	*56s*	*54s*	***1h 18m 21s***	*19s*	*32.3s*	*6m 23s*
Consumer Affairs Dataset	*15.2s*	*21.09s*	*29min 1s*	*29.9s*	*31s*	*7m 45s*

Figure 5.7: CPU Processing time for the various datasets

When comparing Deep Neural Networks and Convolutional Neural Network models, it is clear that CNN requires more time to process data but yields superior assessment metrics.

5.4 Observations

Based on the Performance Metrics, we highlight some of the observations of the sentiment analysis techniques.

- When word embedding is used, the Recurrent Neural Network model provides the best level of dependability; however, its computing time is also the largest. When doing a study of the sentiment of tweets and review datasets, using a Recurrent Neural Network with TF-IDF takes much more time than using other models, and the accuracy of the findings is around only half as good, around 50 percent.

- The DNN model is easy to construct and generates results in a short amount of time — around one minute for the majority of datasets, with the exception of the dataset Twitter, for which the model required twelve minutes to generate the results. Even while the model can be trained in a short amount of time, its accuracy is only satisfactory (between 75% and 80%) across the board in all of the validated datasets, which include tweets and reviews.

- The CNN model may also be trained and evaluated quickly, although it may be a little less fast than DNN in this regard. The model achieves a greater level of accuracy (above 80%) when applied to the tweet data observationss the review dataset.

Chapter 6

Discussion

In this part, we Summarize the acquired findings and how they contribute to answering the Research questions. We also explore aspects that contradict the findings.

The RQ1 is resolved via a systematic literature review as discussed in 5.1. The SLR assisted us in identifying suitable algorithms for our research question. Popular strategies derived from the SLR have aided in identifying the optimal algorithm for addressing the remaining research questions.

The Experiment and the results presented in 5.2 answers the 'RQ2:Which combination of word embedding performs the best with the deep learning model?' The experiment has shown that Before feeding text data (tweets, reviews) into a deep learning model, text data (TF-IDF and word embedding) are converted into a numeric vector. The results produced by TF-IDF are inferior than those produced by word embedding. Moreover, the TF-IDF approach used with the RNN model provides less accurate findings. However, when RNN is combined with word embedding, the outcomes are much improved. Future research may investigate how these and other strategies might be improved to provide even better outcomes.

The Experiment and the results presented in 5.3 answers the 'RQ3:Which Deep Learning model has the best processing time?' Experiments on sentiment analysis included 3 deep learning models (RNN,DNN and CNN). It was discovered that the CNN model provides the optimal balance between processing speed and accuracy of output. Although the RNN model was the most accurate when employed alongside word embedding, the processing time was ten times greater compared to the CNN model. The RNN model is ineffective when combined with the TF-IDF approach, and its much longer processing time does not provide significantly superior results. DNN is a straightforward deep learning model with average processing times and outcomes. Continued studies on deep learning models may concentrate on improving the trade-off between the accuracy of the findings and the processing time.

6.1 Threats to Validity

6.1.1 Internal Validity

Internal validity tells about the correctness of the research.The quality of the data may be the most significant potential internal obstacle for the present investigation. The potential dangers to this thesis's internal validity that may emerge throughout the web scraping phase of the research. While extracting the tweets, sufficient precautions must be taken to ensure that the whole text has been transmitted. If this is not done, the improper sentiment polarity will be formed, which will result in a distorted understanding of the situation.Another risk is that the Algorithms that are used are not the right ones. To deal with this threat, the available data has been looked at, a literature review has been done, and experts has been taken into account to choose algorithms that work well with the data that is currently available. The experiment showed that the algorithms worked the way they were supposed to.

6.1.2 External Validity

The term "external validity" refers to how well the thesis's findings can be used in the real world.In this study, the dataset is taken out from comments on social media. Another risk is that the model is out of date and doesn't fit the current situation. properly. The methods that have been suggested have worked well, so this thesis can be used in other real-world situations.

6.1.3 Conclusion Validity

In order for a conclusion to be valid, the comparisons made in the study must use the right metrics. Metrics were taken into account because they were important to this research and helped get to the conclusion.

Chapter 7

Conclusions and Future Work

7.1 Conclusion

In this study, we present the fundamentals of deep learning models and associated approaches that have been applied to social network data sentiment analysis. We transformed input data using TF-IDF , word embedding before feeding it to deep learning models. DNN, CNN, and RNN architectures were investigated and integrated with TF-IDF, word embedding for sentiment analysis. We ran a series of experiments to test DNN, CNN, and RNN models on various datasets, including tweets and reviews, based on diverse subject matter. Additionally, we addressed relevant studies in the topic. This information, together with the outcomes of our experiments, provides us with a comprehensive understanding of applying deep learning models to sentiment analysis and integrating these models with text preparation approaches.

CNN, DNN, and hybrid techniques were found as the most popular models for sentiment analysis after a study of the relevant literature. Another finding derived from the study was that popular approaches, such as RNN and CNN, are individually evaluated on various datasets in these papers, but there is no comparison analysis of these techniques. In addition, the majority of articles provide outcomes in terms of accuracy without regard for processing time.

The experiments that were carried out as part of this study were planned with the intention of contributing to the filling in of the gaps stated before. We investigated the effects of a variety of datasets, feature extraction methods, and deep learning models, with a particular emphasis on the issue of sentiment analysis. When it comes to doing a sentiment analysis, the findings indicate that it is preferable to use a combination of deep learning algorithms and word embedding rather than TF-IDF. The trials also showed that CNN works better than other models and strikes a decent balance between accuracy and the amount of time it takes for the CPU to execute. In most datasets, the RNN has a reliability that is somewhat better than that of the CNN; nevertheless, its computing time is much greater. The efficiency of

the algorithms is found to be highly dependent on the datasets, which is why it is beneficial to test deep learning techniques with a higher number of datasets in order to cover a wider range of features. This is the final conclusion that can be drawn from the research. All study objectives and aims have been attained, and all research questions have been appropriately addressed and justified.

7.2 Future Work

The primary focus can be on investigating hybrid approaches, which involve the combination of multiple models and techniques in order to improve the accuracy of sentiment classification attained by the single models while concurrently decreasing the amount of computational effort required. The purpose of this is to broaden the scope of the comparative research such that it incorporates not only new methodologies but also new kinds of data. [30,31] As a result, the dependability and processing speed of hybrid models would be assessed using a variety of data, including the status updates, comments, and news found on social media platforms. We will also have the intention of tackling the issue of aspect sentiment analysis to get a more in-depth understanding of user feelings by linking them with certain characteristics or subjects. This is of tremendous importance to a vast number of businesses since it enables them to collect in-depth feedback from customers and, as a result, determine which areas of their goods or services need to be enhanced. [32]

References

[1] K. Jain and S. Kaushal, "A Comparative Study of Machine Learning and Deep Learning Techniques for Sentiment Analysis," 2018 7th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), 2018, pp. 483-487, doi: 10.1109/ICRITO.2018.8748793.

[2] Sohangir, S.; Wang, D.; Pomeranets, A.; Khoshgoftaar, T.M. Big Data: Deep Learning for financial sentiment analysis. J. Big Data 2018, 5, 3

[3] Ain, Q.T.; Ali, M.; Riaz, A.; Noureen, A.; Kamran, M.; Hayat, B.; Rehman, A. Sentiment analysis using deep learning techniques: A review. Int. J. Adv. Comput. Sci. Appl. 2017, 8, 424.

[4] Jangid, H.; Singhal, S.; Shah, R.R.; Zimmermann, R. Aspect-Based Financial Sentiment Analysis using Deep Learning. In Proceedings of the Companion of the The Web Conference 2018 on The Web Conference, Lyon, France, 23–27 April 2018; pp. 1961–1966

[5] H. A. Shehu et al., "Deep Sentiment Analysis: A Case Study on Stemmed Turkish Twitter Data," in IEEE Access, vol. 9, pp. 56836-56854, 2021, doi: 10.1109/ACCESS.2021.3071393.

[6] Kraus, M.; Feuerriegel, S. Sentiment analysis based on rhetorical structure theory: Learning deep neural networks from discourse trees. Expert Syst. Appl. 2019, 118, 65–79.

[7] Singhal, P.; Bhattacharyya, P. Sentiment Analysis and Deep Learning: A Survey; Center for Indian Language Technology, Indian Institute of Technology: Bombay, Indian, 2016.

[8] Erenel, Z.; Adegboye, O.R.; Kusetogullari, H. A New Feature Selection Scheme for Emotion Recognition from Text. Appl. Sci. 2020, 10, 5351. https://doi.org/10.3390/app10155351

[9] Abid, F.; Alam, M.; Yasir, M.; Li, C.J. Sentiment analysis through recurrent variants latterly on convolutional neural network of Twitter. Future Gener. Comput. Syst. 2019, 95, 292–308

[10] Aggarwal, C.C. Neural Networks and Deep Learning;

[11] "Clustering — scikit-learn 0.24.2 documentation." [Online]. Available: https://www.scikit-yb.org/en/latest/api/cluster/elbow.html

[12] Ajay Shrestha and Ausif Mahmood. "Review of deep learning algorithms and architectures". In: IEEE Access 7 (2019), pp. 53040–53065.

[13] Shanshan Yu, Jindian Su, and Da Luo. “Improving bert-based text classification with auxiliary sentence and domain knowledge”. In: IEEE Access 7 (2019), pp. 176600– 176612.

[14] Mickel Hoang, Oskar Alija Bihorac, and Jacobo Rouces. “Aspect-based sentiment analysis using bert”. In: Proceedings of the 22nd Nordic Conference on Computational Linguistics (2019), pp. 187–196.

[15] Lei Zhang, Shuai Wang, and Bing Liu. “Deep learning for sentiment analysis: A survey”. In: Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 8.4 (2018), e1253.

[16] Vineeth G Nair. Getting started with beautiful soup. Packt Publishing Ltd, 2014

[17] S Chris Colbert et al. “The NumPy array: a structure for efficient numerical computation”. In: Computing in Science Engineering. Citeseer. 2011.

[18] Aurélien Géron. Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems. O’Reilly Media, 2019

[19] Zhang, L.; Wang, S.; Liu, B. Deep learning for sentiment analysis: A survey. WIREs Data Min. Knowl. Discov. 2018, 8, e1253.

[20] Britz, D. Recurrent Neural Networks Tutorial, Part 1–Introduction to Rnns. Available online: http://www.wildml.com/2015/09/recurrent-neural-networkstutorial-part-1-introduction-to-rnns/ (accessed on 12 March 2020).

[21] Ruangkanokmas, P.; Achalakul, T.; Akkarajitsakul, K. Deep Belief Networks with Feature Selection for Sentiment Classification. In Proceedings of the 2016 7th International Conference on Intelligent Systems, Modelling and Simulation (ISMS), Bangkok, Thailand, 25–27 January 2016; pp. 9–14.

[22] Vateekul, P.; Koomsubha, T. A study of sentiment analysis using deep learning techniques on Thai Twitter data. In Proceedings of the 2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE), Khon Kaen, Thailand, 13–15 July 2016; pp. 1–6

[23] Ghosh, R.; Ravi, K.; Ravi, V. A novel deep learning architecture for sentiment classification. In Proceedings of the 2016 3rd International Conference on Recent Advances in Information Technology (RAIT), Dhanbad, India, 3–5 March 2016; pp. 511–516.

[24] Bhavitha, B.; Rodrigues, A.P.; Chiplunkar, N.N. Comparative study of machine learning techniques in sentimental analysis. In Proceedings of the 2017 International Conference on Inventive Communication and Computational Technologies (ICICCT), Coimbatore, India, 10–11 March 2017; pp. 216–221

[25] Salas-Zárate, M.P.; Medina-Moreira, J.; Lagos-Ortiz, K.; Luna-Aveiga, H.; Rodriguez-Garcia, M.A.; Valencia-García, R.J.C. Sentiment analysis on tweets about diabetes: An aspect-level approach. Comput. Math. Methods Med. 2017, 2017. [CrossRef] [PubMed]

[26] Zhang, X.; Zheng, X. Comparison of Text Sentiment Analysis Based on Machine Learning. In Proceedings of the 2016 15th International Symposium on Parallel and Distributed Computing (ISPDC), Fuzhou, China, 8–10 July 2016; pp. 230–233.

[27] Malik, V.; Kumar, A. Communication. Sentiment Analysis of Twitter Data Using Naive Bayes Algorithm. Int. J. Recent Innov. Trends Comput. Commun. 2018, 6, 120–125.

[28] Firmino Alves, A.L.; Baptista, C.d.S.; Firmino, A.A.; Oliveira, M.G.d.; Paiva, A.C.D. A Comparison of SVM versus naive-bayes techniques for sentiment analysis in tweets: A case study with the 2013 FIFA confederations cup. In Proceedings of the 20th Brazilian Symposium on Multimedia and the Web, João Pessoa, Brazil, 18–21 November 2014; pp. 123–130.

[29] Medhat, W.; Hassan, A.; Korashy, H. Sentiment analysis algorithms and applications: A survey. Ain Shams Eng. J. 2014, 5, 1093–1113. [CrossRef]

[30] Jain, A.P.; Dandannavar, P. Application of machine learning techniques to sentiment analysis. In Proceedings of the 2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT), Karnataka, India, 21–23 July 2016; pp. 628–632.

[31] Tang, D.; Qin, B.; Liu, T. Deep learning for sentiment analysis: Successful approaches and future challenges. Wiley Interdiscip. Rev. Data Min. Knowl. Discov. 2015, 5, 292–303.

[32] Sharef, N.M.; Zin, H.M.; Nadali, S. Overview and Future Opportunities of Sentiment Analysis Approaches for Big Data. JCS 2016, 12, 153–168

[33] Qian, J.; Niu, Z.; Shi, C. Sentiment Analysis Model on Weather Related Tweets with Deep Neural Network. In Proceedings of the 2018 10th International Conference on Machine Learning and Computing, Macau, China, 26–28 February 2018; pp. 31–35

[34] Roshanfekr, B.; Khadivi, S.; Rahmati, M. Sentiment analysis using deep learning on Persian texts. In Proceedings of the 2017 Iranian Conference on Electrical Engineering (ICEE), Tehran, Iran, 2–4 May 2017; pp. 1503–1508.

[35] Ramadhani, A.M.; Goo, H.S. Twitter sentiment analysis using deep learning methods. In Proceedings of the 2017 7th International Annual Engineering Seminar (InAES), Yogyakarta, Indonesia, 1–2 August 2017; pp. 1–4

[36] Paredes-Valverde, M.A.; Colomo-Palacios, R.; Salas-Zárate, M.D.P.; Valencia-García, R. Sentiment analysis in Spanish for improvement of products and services: A deep learning approach. Sci. Program. 2017, 2017.

[37] Tang, D.; Zhang, M. Deep Learning in Sentiment Analysis. In Deep Learning in Natural Language Processing; Springer: Berlin, Germany, 2018; pp. 219–253.

[38] Araque, O.; Corcuera-Platas, I.; Sanchez-Rada, J.F.; Iglesias, C.A. Enhancing deep learning sentiment analysis with ensemble techniques in social applications. Expert Syst. Appl. 2017, 77, 236–24

[39] Liu, J.; Chang, W.-C.; Wu, Y.; Yang, Y. Deep learning for extreme multi-label text classification. In Proceedings of the 40th International ACM SIGIR

Conference on Research and Development in Information Retrieval, Tokyo, Japan, 7–11 August 2017; pp. 115–124.

[40] “Pandas documentation.” [Online]. Available: https://pandas.pydata.org/docs/

[41] “Scikit-learn.” [Online]. Available: https://scikit-learn.org/stable/

[42] Alharbi, A.S.M.; de Doncker, E. Twitter sentiment analysis with a deep neural network: An enhanced approach using user behavioral information

[43] Jeong, B.; Yoon, J.; Lee, J.-M. Social media mining for product planning: A product opportunity mining approach based on topic modeling and sentiment analysis. Int. J. Inf. Manag. 2019, 48, 280–290.

[44] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén, Experimentation in software engineering, vol. 9783642290442. Berlin: Springer, 2012 ed., 2012.

[45] A. S. Zharmagambetov and A. A. Pak, "Sentiment analysis of a document using deep learning approach and decision trees," 2015 Twelve International Conference on Electronics Computer and Computation (ICECCO), Almaty, Kazakhstan, 2015, pp. 1-4, doi: 10.1109/ICECCO.2015.7416902.

[46] Al-Sallab, A.; Baly, R.; Hajj, H.; Shaban, K.B.; El-Hajj, W.; Badaro, G. Aroma: A recursive deep learning model for opinion mining in arabic as a low resource language. ACM Trans. Asian Low-Resour. Lang. Inf. Process. TALLIP 2017, 16, 1–20.

[47] Z. Hu, J. Hu, W. Ding and X. Zheng, "Review Sentiment Analysis Based on Deep Learning," 2015 IEEE 12th International Conference on e-Business Engineering, Beijing, China, 2015, pp. 87-94, doi: 10.1109/ICEBE.2015.24.

[48] M. -Y. Day and Y. -D. Lin, "Deep Learning for Sentiment Analysis on Google Play Consumer Review," 2017 IEEE International Conference on Information Reuse and Integration (IRI), San Diego, CA, USA, 2017, pp. 382-388, doi: 10.1109/IRI.2017.79.

Faculty of Faculty, Blekinge Institute of Technology, 371 79 Karlskrona, Sweden