A DEEP NEURAL NETWORK APPROACH FOR SENTIMENT ANALYSIS

Liyuan Tan

Bachelor of Engineering

A dissertation submitted in partial fulfilment of the requirements for the degree of

Master of Information Technology and Systems

University of Tasmania

June 2019

Statement of Originality

This thesis contains no material which has been accepted for a degree or diploma by the University or any other institution, except by way of background information and duly acknowledged in the thesis, and to the best of my knowledge and belief no material previously published or written by another person except where due acknowledgment is made in the text of the thesis, nor does the thesis contain any material that infringes copyright.

Signature: _____

Date: _____

Statement of Authority of Access

This thesis may be made available for loan and limited copying in accordance with the *Copyright Act 1968*.

Signature: _____

Date: _____

Abstract

Sentiment analysis can be explained as some approaches or techniques to extract subjective information from human language (Mäntylä et al., 2018). With the advent of the Internet age, the amount of digital data is growing rapidly on the Internet. Today, sentiment analysis is basically equivalent to using computer technology to analyse emotional components in digital data, such as texts. In research, sentiment analysis is always about classifying the polarity of opinions.

Most researchers consider sentiment analysis as a subtask of natural language processing (NLP). Therefore, with the technological breakthroughs in the field of NLP, papers focusing on the field of sentiment analysis have increased year by year, and they have proposed many more effective new approaches for identifying sentiment in the language (Mäntylä et al., 2018).

In this research, the author focuses on text-based data because the sentiment analysis of video and audio is not the mainstream in sentiment analysis research. NLP technology currently focuses on the language itself, while the text has fewer features and noise than video and audio. Thus, it is easier to analyse. Specifically, the data is collected from the shopping site's reviews, social media content, and so on, because these kinds of data are easy to obtain, and the quantity is sufficient. More importantly, they represent the language in life, which makes this research more convincing.

At present, the field of sentiment analysis has encountered bottlenecks. For example, many approaches take a long time but do not get better results. Therefore, this research seeks to apply a novel deep neural network approach to classify the sentiment of the text. This kind of deep neural network called deep extreme learning machine (DELM) was developed in 2015 based on its prototype – extreme learning machine (ELM) (Tang et al., 2015b). It was proven to achieve good results in the field of computer vision (CV).

DELM has rarely been used in the field of NLP since its appearance. Some researchers claimed that DELM has achieved better performance than traditional machine learning algorithms in the field of sentiment analysis (Roul et al., 2017). However, so far no research has systematically compared the performance of DELM against the other deep neural networks in sentiment analysis tasks. Thus, in this

research, the author has performed a series of experiments using multiple datasets to test whether DELM has advantages over the other deep learning approaches in the field of sentiment analysis.

After conducting the experiments, the author concludes that DELM currently does not have advantages on accuracy and training time over the other deep learning approaches in the field of sentiment analysis. In detail, the author uses LSTM, DBN, and DELM to classify the sentiment of texts in six datasets. The experiments are implemented on the same hardware and operating system. The experimental results in this research show that DELM takes the longest time on five datasets and the second long time on one dataset in training. Further, DELM achieves the highest accuracy on one dataset, the second high accuracy on two datasets and the lowest accuracy on three datasets.

There are two main reasons for the experimental results. First, currently, the word embedding technology is tied to the LSTM model, but it is not compatible with the DBN model and the DELM model. This restricts the performance of DELM in this research. Second, DELM runs in the MATLAB environment in this research. MATLAB is a memory-intensive program, so the training time of DELM in the experiments is lengthened. However, in the author's opinion, DELM still has the potential to become an effective sentiment classification approach because, in the results, it has close accuracies compared to mainstream approaches under the influence of many unfavorable factors.

Acknowledgments

I am very grateful for the assistance provided by my supervisor, Dr Shuxiang Xu. His assistance in carrying out this research and completing this thesis is invaluable. Without him and his support and suggestions, I would not have been able to undertake this research on time.

I would also like to thank Dr Robert Ollington, for providing help during this year.

I would also like to thank albertbup, GitHub, for the provision of the source code of deep belief network.

I would also like to thank Prof. Guangbin Huang, Nanyang Technological University, Singapore, for the provision of the source code of deep extreme learning machine.

I would also like to thank Kaggle for making the datasets available.

Thanks also to my parents and my school mates, Zeyu Zhang, Ling Cai, and Tianyi Shan, for their love and encouragement.

Table of Contents

List of Figures

List of Tables

Chapter 1: Introduction

This chapter outlines the introduction (section 1.1) and background (section 1.2) of the research, the research objective, and the research question (section 1.3). Section 1.4 indicates the contributions of this research in brief. Finally, section 1.5 includes an outline of the remaining chapters of the thesis.

1.1 INTRODUCTION

Sentiment analysis can be explained as some approaches or techniques to extract subjective information from human language (Mäntylä et al., 2018). With the advent of the Internet age, the amount of digital data is growing rapidly on the Internet. Today, sentiment analysis is basically equivalent to using computer technology to analyse emotional components in digital data, such as texts. In research, sentiment analysis is always about classifying the polarity of opinions.

Most researchers consider sentiment analysis as a subtask of natural language processing (NLP). Therefore, with the technological breakthroughs in the field of NLP, papers focusing on the field of sentiment analysis have increased year by year, and they have proposed many more effective new approaches for identifying sentiment in the language (Mäntylä et al., 2018).

In this research, the author focuses on text-based data because the sentiment analysis of video and audio is not the mainstream in sentiment analysis research. NLP technology currently focuses on the language itself, while the text has fewer features and noise than video and audio. Thus, it is easier to analyse. Specifically, the data is collected from the shopping site's reviews, social media content, and so on, because these kinds of data are easy to obtain, and the quantity is sufficient. More importantly, they represent the language in life, which makes this research more convincing.

At present, the field of sentiment analysis has encountered bottlenecks. For example, many approaches take a long time but do not get better results. Therefore, this research seeks to apply a novel deep neural network approach to classify the sentiment of the text. This kind of deep neural network called deep extreme learning machine (DELM) was developed in 2015 based on its prototype – extreme learning machine (ELM) (Tang et al., 2015a). It was proven to achieve good results in the field of computer vision (CV).

DELM has rarely been used in the field of NLP since its appearance. Some researchers claimed that DELM has achieved better performance than traditional machine learning algorithms in the field of sentiment analysis (Roul et al., 2017). However, so far no research has systematically compared the performance of DELM against the other deep neural networks in sentiment analysis tasks. Thus, in this research, the author has performed a series of experiments using multiple datasets to

test whether DELM has advantages over the other deep learning approaches in the field of sentiment analysis.

1.2 BACKGROUND

Papers about sentiment analysis appeared in the 1940s but have been changed and developed since that time. Public opinions of WWII were the research objects of sentiment analysis at the very beginning. However, the value of these studies is limited because they are not under the modern NLP framework. In the mid-90s, NLP based on computer science came out, and modern sentiment analysis was born at that time. At the beginning of the 21st century, sentiment classification used machine learning and achieved 74% accuracy for online reviews. This trend has led researchers in computer science to continue to contribute more excellent algorithms, and even linguistic researchers have joined many collaborative projects (Mäntylä et al., 2018).

Machine learning has always been the protagonist in the field of modern sentiment analysis research. Since the beginning of the 21st century, almost all mainstream machine learning algorithms have been applied to this research field. According to Qazi et al. (2017), Naïve Bayes classification, maximum entropy classification and support vector machines (SVM) were used to classify movie reviews in 2002. SVM achieved the best performance as 82.9% accuracy. In 2005, unsupervised NLP techniques were used to identify the polarity of Amazon reviews and achieved 79% accuracy. In 2009, the n-gram model was first used to classify travel reviews on Yahoo.com and got 80% accuracy.

In 2011, some researchers proposed an approach for sentiment analysis using deep neural networks. Surprisingly, this approach achieved state-of-the-art performance on a data set based on 22 different domains. They used Restricted Boltzmann Machines (RBMs) and Auto-encoder which both were popular deep architectures for classifying (Glorot et al., 2011). Then, more researchers began to use deep neural network algorithms for text classification tasks.

Recently, a novel deep neural network approach has entered the field of vision of many researchers. This kind of deep neural network called deep extreme learning machine (DELM) is not really a new approach. In 2006, Guang-Bin Huang, a professor at Nanyang Technological University, announced its structure to the public. At that time, it was called an extreme learning machine (ELM) because it belongs to single-

hidden layer feedforward neural networks (SLFNs) (Huang et al., 2006). In 2015, researchers updated this kind of neural network. The auto-encoder technique was applied, which made ELM become DELM. This deep neural network approach has become a hot research topic because it is more efficient than other deep neural network algorithms in some applications. Taking the image classification task as an example, the training time of DELM is significantly lower than other mainstream deep neural network algorithms while maintaining the same accuracy (Tang et al., 2015b). However, DELM has rarely been used in the field of NLP since its appearance.

1.3 RESEARCH OBJECTIVE AND QUESTION

So far no research has systematically compared the performance of DELM against the other deep neural networks in sentiment analysis tasks. Thus, in this research the author aims at performing a series of experiments using multiple datasets to test whether DELM has advantages over the other deep learning approaches in the field of sentiment analysis.

The primary research objective for this thesis is the following:

- To establish whether DELM achieves better accuracy and takes less training time when conducting sentiment classification tasks on six text-based data sets in different domains, in comparison with LSTM and DBN.

In addition to the above objective, there are the following sub-objectives:

- To identify and select suitable text-based datasets in different domains for sentiment classification tasks based on deep neural networks.
- To identify and select the most suitable feature selection approach for sentiment classification tasks based on deep neural networks.
- To evaluate the quality of three deep neural networks used to classify text-based data.

The primary research question for this thesis is "Does DELM have advantages on accuracy and training time over other deep neural network algorithms in sentiment analysis?"

1.4 CONTRIBUTIONS

The contributions made by this research are as follows.

- Introducing suitable datasets for sentiment classification tasks based on deep neural networks.
- Introducing an effective feature selection approach for text-based sentiment classification tasks based on deep neural networks.
- Demonstrating the strengths and weaknesses of DELM in text-based sentiment classification tasks based on the results of experiments.

1.5 THESIS OUTLINE

The following is a structure of the chapters in this thesis:

The next chapter is a literature review of sentiment analysis. It investigates the basic concepts of sentiment analysis, development trends, and main technical components. The main length covers related techniques for feature selection and sentiment classification. Several deep neural networks including LSTM, DBN, DELM used in this research are also introduced in detail.

The third chapter demonstrates the methodology, to explain how this research was conducted to answer the research question. The process of sentiment analysis in this research includes four distinct phases, as follows: (1) Datasets Selection, (2) Text-based Data Pre-processing, (3) Deep Neural Network Classification and (4) Performance Evaluation.

The fourth chapter collects the results by doing the experiment based on the process mentioned in the previous chapter. This chapter discusses the findings of the research. Furthermore, it evaluates the methodology and discusses how to improve.

The conclusion chapter indicates the contributions of this thesis. Moreover, this chapter also provides the limitations of this research and then suggests directions for further work.

Chapter 2: Literature Review

This chapter begins with a background (section 2.1) about sentiment analysis, including its definition, demand, application, importance and process, and reviews literature on the following topics: feature selection (section 2.2) which investigates approaches of sorting text-based data and converting them to a format that the computer can understand; sentiment classification (section 2.3) which detects approaches of classifying data by different algorithms; and deep neural network approach (section 2.4) collects some novel sentiment classification approaches are used in this research and needs specific explanation. Section 2.5 highlights the implications of the literature and develops the conceptual framework for the research.

2.1 BACKGROUND

A significant purpose of exchanging information is figuring out what others think. In the age of the Internet, people's desire for emotional communication has promoted the rise of online interactive platforms and social media. Simultaneously, people's interaction on the Internet produces a lot of data based on text mixed with opinion and sentiment. Thus, the field of opinion mining and sentiment analysis deserves an in-depth study because there is a considerable amount of valuable information hidden in massive text (Pang and Lee, 2008).

2.1.1 Definition

According to Liu (2015), a professor at the University of Illinois, “sentiment analysis, also called opinion mining, is the field of study that analyses people’s opinions, sentiments, appraisals, attitudes, and emotions toward entities and their attributes expressed in written text.”

2.1.2 Demand

People are always curious about what others think, which can influence their behaviours and decision-making process (Pang and Lee, 2008). Sentiment analysis is important for companies and organizations because they need to know what their customers think about the service and products. The government is also expected to detect public opinions accurately and fast because it is closely related to the formulation of policies and the response to social and economic emergencies. The demand for information about sentiment exists in large numbers in society.

However, in the past, this demand was not easily met because of technical limitations. For example, individuals could only get views of their relatives and friends in daily life. If more information should be collected, complicated qualitative researches needed to be done such as arranging interviews, doing surveys and organizing focus groups. It took a lot of time and might achieve an unsatisfactory result because researchers might not have enough experience, research approaches might not be scientific enough, and there were mistakes in detail in the research process. For organizations and governments, although they were able to allocate sufficient social resources to get the information they want, such as through newspapers, radio, and television. However, when facing a large amount of data, collecting is easy, but the analysis is difficult (Liu, 2015).

2.1.3 Applications

In recent years, social media that contains a remarkable amount of text that was unimaginable in the past is expanding, such as Twitter, Facebook, Weibo, and WeChat. Thus, people can get advice from new friends on social media about their decision-making. Moreover, people can take others' opinions as references easily because most people are willing to share their values and experience on social media. Companies and organizations can also judge whether their services and products are accepted by the market based on users' messages under their official social media accounts. With the consent of companies and customers, the government can also monitor the public sentiment by analysing the emotions of people embedded in the text (Liu, 2015).

In addition to the applications mentioned above, the researchers are looking for new application scenarios for sentiment analysis in every possible domain. Due to a huge volume of opinion text on websites and social media, people cannot summary the general sentiment by just reading them. Thus, automated sentiment analysis systems should be introduced to business and research purposes, which is explained in detail in the next subsections. In universities and large IT companies like Google, Amazon, Microsoft, Oracle and SAP, these sentiment analysis systems, also called "Opinion Parser", are applied in different projects, such as word processing, video processing and financial analysis (Liu, 2015). Figure 2-1 shows the applications of sentiment analysis described in section 2.1.

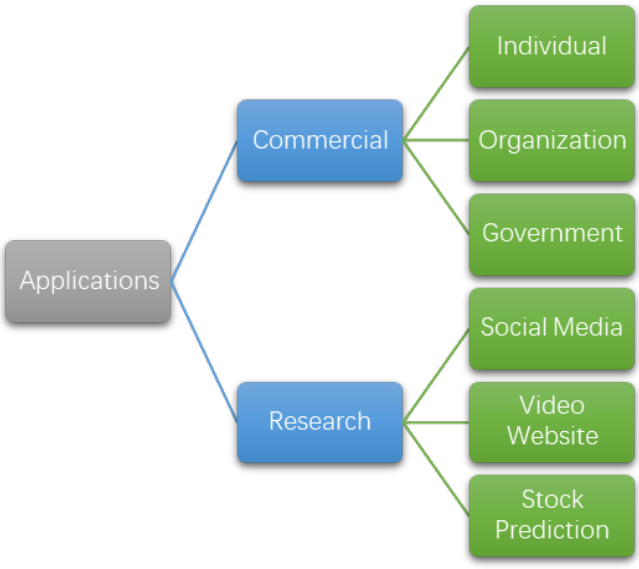

Figure 2-1 Applications of Sentiment Analysis

Research Applications: Social Media

For example, Kiritchenko et al. (2014) developed "a state-of-the-art sentiment analysis system" dedicated to analysing short informal texts on social media. They were concerned with the massive expression of colloquialism in text form. Tweets in some daily topics such as movies, reviews of products, politicians were the objects of the research. They had their unique sentiment lexicons which provided functions to deal with common misspellings, elongations, and abbreviations. Furthermore, the sentiment lexicons were taught to recognize polarity of every word, such as positive, negative and neutral.

However, the researchers came up with a novel idea to modify the words' polarity. They made two sores for one word's polarity because words could carry different meanings in negated contexts and affirmative contexts. This detailed pre-treatment improved the accuracy of identification, which brought a competitive advantage for the sentiment lexicons. There was another contribution of this research group which was about expanding sentiment analysis researchers' views. The researchers indicated that in the past decade, sentiment analysis was limited in few approaches such as "detecting subjective and objective sentences" and "classifying sentences as positive, negative, or neutral", labelling words had become the main job of sentiment analysis.

Critique

Although the experiment design was very clever, there were some problems embedded in the process of the experiment. First, manual annotation cost time and money, which was not suitable for big data because of the lack of efficiency. Second, the common way only determined the polarity of subjective text, which ignored the role of objective text in sentiment analysis. The researchers ranked the normal way and the fixed way using their sentiment analysis systems by comparing data about the performance. They reminded other researchers that traditional approaches for sentiment analysis need to be improved and words could be annotated automatically because they detected intentions behind 135 million tweets in 11 hours with 50 machines.

Relevance

This experiment inspired the author that sentiment analysis requires more approaches to get more comprehensive results. This is why the author uses three different sentiment classification algorithms in the research object. This experiment had limitations in feature selection so that the author uses better feature selection approaches in this research. This is related to the second research sub-objective.

Research Applications: Video Website

Another example is about sentiment analysis application in video processing. Some researchers found that currently, most sentiment analyses were restricted to text-based sources, which was not sufficient for research. People were not satisfied with communicating and expressing their opinions just in text because videos and images could be more attractive, and easy to read. Therefore, diverse modalities should be implemented when researching opinion mining. The main part of the experiment was consistent with normal sentiment analysis. First, they extracted information from YouTube who was willing to offer the dataset. Then, they discussed the features they were concerned about. Next, they employed classifiers based on machine learning for the classification part. Finally, they found the extreme learning machine (ELM) had significant advantages in performance (The detailed process for ELM is demonstrated in section 2.4 Deep Neural Network Approach).

The highlight of this research was building a multimodal sentiment system by analysing different types of data. Except for textual data, video and audio were also taken into consideration. For video, facial expressions were the main objects which could be divided into seven types: anger, sadness, surprise, fear, disgust, joy and contempt. Some common algorithms like k-nearest-neighbours, Bayesian networks, hidden Markov models (HMM), and artificial neural networks (ANN) were used to get the features (Algorithms for sentiment classification are demonstrated in detail in section 2.3 Sentiment Classification and section 2.4 Deep Neural Network Approach). For audio, they focused on the pitch, the intensity of utterance, bandwidth, and duration. Gaussian mixture model (GMM) was applied to get 98% accuracy in classification (Algorithms for sentiment classification are demonstrated in detail in section 2.3 Sentiment Classification). Then, researchers started the heart of multimodal sentiment analysis engine -- multimodal fusion which could be divided into feature-level fusion and decision-level fusion. Feature-level fusion made a feature vector combined with inputs from video, audio, and text, and decision-level fusion calculated

the final outputs from separate outputs when confirming dealing with three kinds of inputs independently. The result reminded other researchers that sentiment analysis could be applied in a variety of media because the performance of the mixed sentiment analysis system was better than the text-based one (Poria et al., 2016).

Critique

Although the experiment was conducted under a feasible framework, and it achieved the state-of-the-art in video sentiment detection. The researchers did not improve classifying ability with facial features, which are important features in videos. Moreover, the feature extraction step was cumbersome in this experiment, which may cause errors when applied to actual videos.

Relevance

This experiment proved that the DELM approach could be used in sentiment analysis, and it may achieve good results in different datasets in this research. This is the premise of the research object of this research.

Research Applications: Stock Prediction

The last example is related to finance, which was a research point combining computer science and economics. Some researchers found that sentiment analysis could be applied to predict the stock in the real world because the theory of economics told people that emotions might affect individual behaviour and judgment and the fluctuation of stock price had a strong correlation with people's decision-making. However, individual emotions were unstable, and it was difficult to detect individual emotions accurately. Fortunately, with the rise of social networks and the increased willingness of people to express their views on social networks, getting data about the public mood became much easier. Thus, the researchers decided to look for the relationship between emotions embedded in large-scale tweets and the value of the Dow Jones Industrial Average (DJIA). Two “Opinion Parsers” were used to divide emotions into six categories -- Calm, Alert, Sure, Vital, Kind, and Happy. Then, professional tools were used to extract tweets in a specific period which would be matched with the value of the DJIA at the same time. Next, the assumptions were made based on the results analysed by deep neural networks which were an important reference for stock price forecasting. At last, researchers compared the prediction and the reality, which proved that sentiment analysis had a significant impact on the Dow

Jones index because when the mean of price reduced more than 6%, the accuracy of prediction was 86.7%. Thus, public mood had a statistically significant correlation with the Dow Jones index, which meant that sentiment analysis could be applied to the field of financial analysis (Bollen et al., 2011).

Critique

Although the experiment discovered the connection between public sentiment and stock index. The researchers believed that the positive and negative labels of information were not equivalent to the rise and fall of stock prices. This also led to that the accuracy of the experiment could only be used as a reference, but it could not be recognized as a fact.

Relevance

This experiment indicated that the label had a huge impact on results, and the same approach performed very differently on different datasets. This led me to the idea of adding more datasets to this research, and they should include multiple labels other than positive and negative. This is related to the first research sub-objective.

2.1.4 Importance

Sentiment analysis has received a lot of attention in recent years, and its importance has also been discovered. People from different industries in different regions have recognized the value of sentiment analysis and be optimistic about the prospects of sentiment analysis. According to an interview on 2000 American adults, 81 percent of people who used the web had experience of looking for what they need on online shopping websites. Moreover, most people admitted that the review of others on the website could have a huge impact on their decision-making. When choosing goods and services, they preferred those with 5-star than 4-star-rated items. Thus, people need sentiment analysis to help them improve the efficiency of online shopping. This was not the only motivation for people to seek for sentiment analysis. Another interview on 2500 American adults showed that consumption of political information was also a key point. It took time for people to achieve the correct perspectives of their community. However, sentiment analysis could do it for them. 28% said that they would share their views online and 27% indicated that they would find useful information online. Thus, sentiment analysis also improved the efficiency of public participation in political (Abbas et al., 2018).

Both demand and applications of sentiment analysis increased rapidly in recent years as mentioned above. Therefore, the public and governments should pay more attention to the importance of sentiment analysis because with the development of social media and e-commerce, the opinion of the public plays a more significant role in the world (Ravi and Ravi, 2015).

2.1.5 Process of Sentiment Analysis

The process of sentiment analysis is normally the same, shown in Figure 2-2. First, the sentiment in the real world should be extracted by sentiment identification, normally it should be words or phrases in human language. Second, the features should be selected which would be used in the next step – sentiment classification. Sentiment classification is the most important one because this step uses several different techniques to identify sentiments. Finally, researchers get sentiment polarity from the algorithms (Medhat et al., 2014). In section 2.2 Feature Selection, section 2.3 Sentiment Classification and section 2.4 Deep Neural Network Approach, the focus is on techniques related to feature selection, sentiment classification and sentiment classification approaches based on deep neural networks.

Figure 2-2 Process of Sentiment Analysis (Medhat et al., 2014)

2.2 FEATURE SELECTION

In section 2.2, there is a collection for existing feature selection techniques used in sentiment analysis. All the descriptions based on actual applications in published journals and conference papers (Medhat et al., 2014).

There are currently four features worth being extracting in feature selection. First, the presence and frequency of terms are significant because they clearly show whether the words are used and how many times they are used. Then, parts of speech play an important role since it helps researchers find adjectives which are keys of sentiment direction. Next, opinion words and phrases are what researchers care about because they directly demonstrate the sentiment, and they are easily recognized. Finally, negations should be watched because it can immediately change the opinion

of the expression, such as the difference between ‘happy’ and ‘not happy’ (Roiger, 2017). Figure 2-3 shows techniques of feature selection described in section 2.2 Feature Selection.

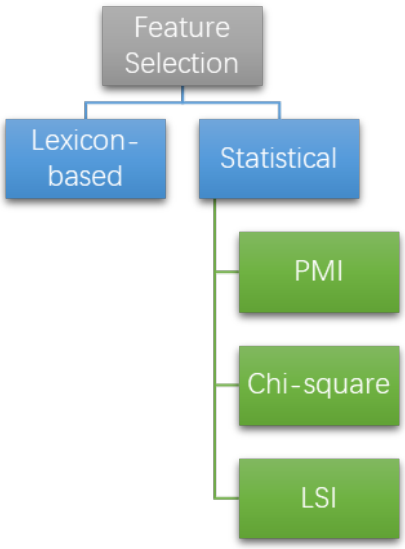

Figure 2-3 Feature Selection Techniques (Medhat et al., 2014)

2.2.1 Lexicon-based Approach

Lexicon-based feature selection is one of the most commonly used approaches in sentiment analysis. Unlike statistical approaches, this approach focuses on the application of linguistics in sentiment analysis. Linguistic experts worked with computer experts to create a dictionary that could distinguish the sentiment trends of words which were named ‘The Semantic Orientation Calculator (SO-CAL)’. Polarity and strength of words were signed in detail in SO-CAL to make extraction from texts easier and more convenient. The work started from discovering a list of words by finding ‘seed words’ which could be adjectives, verbs, nouns, or adverbs embedded with the sentiment. The first step was calculating the sentiment value of ‘seed words’ and ranking them by testing if the words had a prior polarity which means the semantic orientation not related to the context. Second, the researchers found that the frequency of ‘seed words’, especially adjectives, should also be taken into consideration because they always caught strong polarity such as positive and negative, which could affect the sentiment degree of texts significantly. Third, some nouns and verbs were difficult to handle because they carry both neutral and non-neutral connotations. For these words, the rank should be made based on their most frequent usage in daily life. For

example, ‘fabricate’ always had a negative meaning in context, so it was ranked ‘-2’. Then, a series of complex linguistic approaches were also used to make the detail of SO-CAL more reliable. Finally, when comparing with some existing dictionaries about sentiment, the accuracy of the SO-CAL dictionary was significantly better (Taboada et al., 2011). The result of the experiment is shown in the following table.

Table 2-1 Comparison of Performance Using Different Dictionaries (Taboada et al., 2011)

Dictionary	Percent correct		
	Movie	**Camera**	**Overall**
Google	66.31	61.25	62.98
Maryland	67.42	59.46	62.65
GI	64.21	72.33	68.02
SentiWordNet	61.89	67.00	65.02
Subjectivity	65.42	77.21	72.04
SO-CAL	76.37	80.16	78.74

Critique

However, the manually built dictionary still had some disadvantages such as taking too much time to edit and modify, and detailed problems could not be found easily.

Relevance

This experiment showed that the lexicon-based approach was not suitable for large data volumes. Therefore, the author uses a more efficient feature selection approach in this research. This is related to the second research sub-objective.

2.2.2 Pointwise Mutual Information

Because of human could not exhaustively get sentiment rank of all vocabulary when applied in different contexts, statistical approaches in feature selection were introduced to more mathematical sentiment analysis. These approaches treat

documents as a bag of words or a string and deal with them in sequences automatically based on special mathematical models (Medhat et al., 2014).

One of the most popular statistical approaches is Pointwise Mutual Information (PMI). As the name of this approach identified, it pays more attention to the mutual information between features and classes when extracting information from texts. Mutual information (MI) can be considered as the information overlap between two random variables. If the two variables are x and y, their probabilities are $p(x)$ and $p(y)$, and the joint probability is $p(x, y)$, The MI is calculated by the following formula.

$$I(X;Y) = \sum_{x,y} p(x,y) \ln \frac{p(x,y)}{p(x)p(y)}$$

In brief, this approach makes researchers understand the gap between the level of co-occurrence of events occurring $p(x, y)$ and what are expected to occur when two variables are independent $p(x)p(y)$. MI can be positive or negative, but in most cases, it is positive (Bouma, 2009).

When dealing with feature selection in the former sentiment analysis experiment, PMI could be used to compare two 'seed words', and their similarity could be calculated based on the contextual entropy model. This promoted the expansion of the 'seed word', which meant discovering new words that were similar to the 'seed word'. In actual experiments, both 'seed words' and extension words could be applied to sentiment classification (Medhat et al., 2014).

Critique

In the experiment, the approach relied on association scores. If researchers had tried to fix the problem by applying a new reference distribution, normalization strategies need to change. Since new normalized measures might force computers to run longer, this approach was computationally expensive.

Relevance

This experiment showed that PMI might not suitable for this research because feature selection should not take too much time. Otherwise, this will cause the experiment in this research to take too long and delay sentiment classification tasks.

Therefore, the author uses a more efficient feature selection approach. This is related to the second research sub-objective.

2.2.3 Chi-square

There is also another statistical approach that can calculate the correlation between different terms – Chi-square. This is a numerical test that can detect the gap among the distribution of what is expected based on the independence of features and classes. The chi-square value is calculated by the following formula.

$$chi - square_{metric} = t(t_p, (t_p + f_p)P_{pos}) + t(f_n, (f_n + t_n)P_{pos}) + t(f_p, (t_p + f_p)P_{neg}) + t(t_n, (t_n + f_n)P_{neg})$$

t_p represents true positives, f_p represents false positives, t_n represents true negatives, f_n represents false negatives, P_{pos} represents the probability of the number of positive cases, P_{neg} represents the probability of the number of negative cases. The following process is identifying the hypothesis, making the plan, dealing with sample data and concluding the result, which is similar to the common process of quantitative research (Thaseen and Kumar, 2017).

When using this approach for feature selection in the former sentiment analysis experiment, the selection could be more precise because a reliable feedback feature could be introduced into the process. When facing complex feature types, the combination of chi-square and SVM greatly improved the accuracy of the classification. Moreover, it reduced the number of characteristic noises and reduced over-fitting, thereby improving sentiment analysis (Medhat et al., 2014).

Critique

In the experiment, the approach needed the hypothesis, the analysis plan, and sample data to get the results. It meant that when applying this approach, researchers should be familiar with the datasets. However, this is a condition that cannot be guaranteed in sentiment analysis research. Thus, this approach has limited application range.

Relevance

This experiment shows that the chi-square approach may not suitable for this research because of certain preconditions. Therefore, the author uses a more adaptable feature selection approach. This is related to the second research sub-objective.

2.2.4 Latent Semantic Indexing

Latent semantic indexing (LSI) is a well-known statistical approach that can extract useful information and detect the latent semantics by reducing the dimensionality of the original documents. The axis-system is built from the text space by the approach when keeping the semantic structure and main information at the same time. One of the most important purposes of LSI is making the number of dimensions smaller, but it should still be more reliable and stronger than a single term. LSI can build a semantic vector space from the matrix by dropping dimensions related to noise based on the variability when choosing words. The main process of LSI can be explained by the following formula.

$$\hat{q} = q^T U_K \sum\nolimits_K^{-1}$$

$\hat{q}$ represents the query vector which is the result after processing LSI. q represents the original matrix, U represents the matrix of terms vectors in all materials, $\sum_K^{-1}$ is the diagonal matrix of singular values (Song and Park, 2009).

Critique

As shown above, LSI cannot cover features at low levels which is its disadvantage.

Relevance

This technique mainly gets high-level features. However, in sentiment analysis research, features are mainly at a low level. Therefore, the author uses an alternative feature selection approach. This is related to the second research sub-objective.

2.2.5 Summary of Feature Selection

Feature selection is just the pre-processing of data, but it is still an important one. However, whether using lexicon-based approaches or statistical approaches, the accuracy and efficiency of experiments cannot reach a high level. Human language has been developed for thousands of years, so it is difficult to accurately define the sentiment categories based on a few simple parameters, let alone the differences

between culture and countries. However, progress is happening every day. Researchers have developed many different techniques from the old one like Hidden Markov Model to the new one related to deep neural networks which can be more effective (Parlar et al., 2018).

2.3 SENTIMENT CLASSIFICATION

Sentiment classification is the key step in sentiment analysis research. All the approaches in this step can be classified as a machine learning approach, a lexicon-based approach, and a mixed approach. First, machine learning (ML) approaches focus on machine learning algorithms which are commonly robust and adaptable. It can also be classified as supervised and unsupervised learning approaches. Supervised learning approaches use a considerable number of labeled texts for training. In contrast, unsupervised learning approaches are applied when there are not enough labeled texts. Second, lexicon-based approaches rely on statistical approaches to discover sentiment types. It can also be classified as dictionary-based approaches that need dictionaries including 'seed words' and other related words, and corpus-based approaches that need large corpus including a list of 'seed words' embedded sentiment and other related words. Third, mix approaches are also common because they have advantages from both approaches (Medhat et al., 2014). Figure 2-4 shows techniques of sentiment classification described in section 2.3 Sentiment Classification.

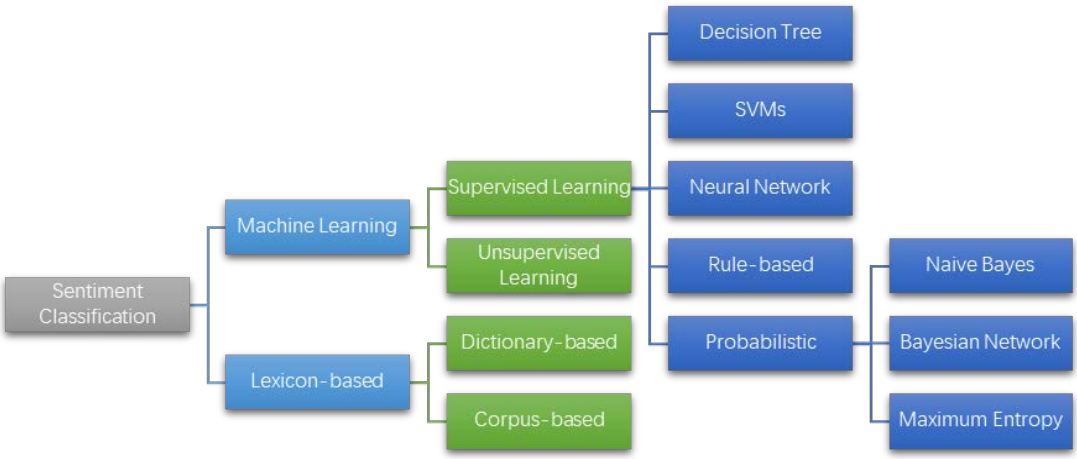

Figure 2-4 Sentiment Classification Techniques (Medhat et al., 2014)

2.3.1 Decision Tree Classifier

Machine learning approaches are all about machine learning algorithms which are commonly used in text classification. The process of text classification can be described briefly as following. First, in feature selection, some data records for training with labels are made, and each one can be considered as a class. Then, the classification models based on algorithms establish a strong connection with the features of labelled classes. Next, the models can predict the features of other given classes with no label. Finally, results can be generated by the models about whether the given class has particular features or the probability of having these features. Supervised learning approaches use many labelled data objects with various supervised classifiers for training (Medhat et al., 2014). These classifiers are introduced starting from section 2.3.1 Decision Tree Classifier.

The decision tree algorithm is essentially a recursive algorithm based on a depth-first greedy approach or breadth-first approach. It is just like a tree that has root nodes, internal nodes, and leaf nodes. Data objects split in each internal node with the best decisions. The first part of the classification is 'building the tree' which starts from the top. Data is recursively sent to different classes. The second part of classification is 'pruning the tree' which starts from the bottom. It deals with noise in detail about the training data. The decision tree classifier is widely used in text mining because it is cheap and easy to understand. It can be operated in parallel, which saves time and is suitable for a large number of data objects. It adapts to the underlying computer architecture and has a longer history than other classifiers. However, its performance shows instability in recent years because of too many variants of the algorithm. Speed and accuracy vary greatly in different application scenarios. Thus, researchers are developing other classifier algorithms for better results (Anyanwu and Shiva, 2009).

Critique

In sentiment classification, training data is text documents, and the classification principle is the existence or not existence of particular words or phrases. The traditional one is simple and already not applicable to current research, so it needs improvement in detail.

Relevance

This technique shows that the decision-tree-based approach is computationally expensive as a sentiment classifier. Therefore, the author tries a deep neural network to classify sentiment in texts. This is related to the establishment of the research objective and the research question.

2.3.2 Support Vector Machines Classifier

Support vector machines (SVMs) are important algorithms in text mining as mainly supervised classifiers in machine learning. The output of simple SVM can be calculated by the following formula. p represents the output, $\bar{A}$ represents a vector of coefficients, $\bar{X}$ represents word frequency in the data object.

$$p = \bar{A}.\bar{X} + b$$

Normally, SVMs learn functions at first. Then, they use linear separators to generate output data from input data which is normally vectors. The classification in SVMs is a hyperplane in the feature space. There are always many hyperplanes between two classes when the largest distance of any of the data points appears, the best separation occurs.

In sentiment classification, SVMs get advantages because data based on text is suitable for SVMs. Most features in the text are relevant so that they can be divided into linearly separable categories. Some researchers use One-versus-All SVM and Single-Machine Multi-class SVM for classification (Yu and Kim, 2012). SVMs are frequently used in machine learning classification tasks because of their stability and ease of use (AL-AMRANI et al., 2018).

Critique

The binary classification is always simple and easy to understand, but in sentiment analysis, the classification is much more complicated. The improved SVMs can consider more variables and perform more internal classifications to get the desired results.

Relevance

This technique shows that SVMs are generally suitable as sentiment classifiers. Therefore, the author uses it as an important reference to select appropriate deep neural networks. However, this research mainly tests deep neural networks. Thus, SVMs are

not applied to experiments. This is related to the establishment of the research objective and the research question.

2.3.3 Neural Network Classifier

Neural networks are special algorithms as popular supervised classifiers in machine learning. Commonly, neural networks are formed from a large number of neurons designed as basic units. Similar to the human brain, when using neural network algorithms, the input generally enters the system from particular neurons. Then, a set of weights identifies the relationship among every neuron accepted inputs from one particular function. Finally, the system makes the prediction just like the process in SVMs. In neural networks with multilayers, non-linear boundaries are used to make enclosed regions for different classes.

In sentiment classification, neural networks have their special roles. The overall sentiment is hard to classify because encoding the intrinsic relations between sentences cannot be completed by simply labelling some words. When using typical sentiment classification techniques like SVMs, relations between sentences are normally not taken into consideration, which causes mistakes in sentiment identification. Neural networks focus on the principle of compositionality which can identify sentiment embedded in longer texts. In the early sentiment analysis research, neural networks did not attract researchers' attention. However, in recent years, with the increasing demand in long text analysis, deep neural networks that are developed from basic neural networks perform better than other algorithms, and there is a statistically significant difference (Tang et al., 2015a). The author describes deep neural networks in detail in section 2.4 Deep Neural Network Approach.

Critique

In sentiment classification tasks, neural networks can distinguish nonlinear features in text data like SVMs. In the early years, simple neural networks could achieve good results in a real application such as recommendation systems. However, when dealing with big data, simple neural networks have the poor fitting ability when facing a large number of features.

Relevance

This technique shows that neural networks are a good basic sentiment classifier. Therefore, the author tries its upgraded version -- deep neural network to classify

sentiment in texts. This is related to the establishment of the research objective and the research question.

2.3.4 Rule-based Classifier

Rule-based classification algorithms make class labels based on models established by rules. In daily life, datasets are always unevenly distributed, sometimes may have omissions and errors. Network latency, outdated sources and errors in sampling all can be the reason. Thus, if researchers generate rules by setting different rule weights and using some operators in terms of interval and probability distribution of uncertain data, the prediction is corrected, and the performance cannot be worse due to the high uncertainty of data. However, some pre-processing approaches of instances can also solve data imbalances. The advantage of the rule-based classifier is that it can build combination algorithms based on support and confidence. Moreover, encoding rules on the feature space can also be implemented, which allows overlaps in the decision (Fernández et al., 2008).

In sentiment classification, the rule-based classifier is useful because, in many datasets, the data is not in balance. For example, comments under one product on Amazon are not distributed on average from one star to five stars. Thus, when identifying the sentiment of all the comments, rules should be established to make data more balanced and easy to deal with.

Critique

In sentiment classification, the rule-based classifier is not used frequently because it takes too much time to design. Before using rule-based classifiers, rules should be established, so that researchers should be familiar with the datasets. Because of its characteristics, its mobility becomes poor.

Relevance

This technique shows that rule-based classifiers can be used as an auxiliary function when data in the datasets is not balanced. Therefore, the author does not use it in experiments because data in datasets is balanced in this research. This is related to the establishment of the research objective and the research question.

2.3.5 Naı̈ve Bayes Classifier

Probabilistic classifiers use the probability of particular terms for prediction. The model may have different classes that own different probabilities for different terms. Generally, there are three famous probabilistic classifiers.

The Naı̈ve Bayes classifier can be easily found in research about text mining in machine learning because it is convenient to use, and the theory is easy to understand. Among all the classifiers in machine learning, the accuracy of Naı̈ve Bayes may not be the highest. However, it makes every parameter contribute equally to the final result with independence. Calculation using Naı̈ve Bayes can be fast and reliable.

When using Naı̈ve Bayes classifier for sentiment classification, features are extracted from a bag of words without information about the roles of words in their sentences or paragraphs. Bayes Theorem is used to find the probability of a particular feature belonging to a particular label, which can be explained by the following formula.

$$P(label|features) = \frac{P(label) * P(features|label)}{P(features)}$$

$P(label)$ represents the probability of a feature belongs to the particular label, $P(features)$ represents the probability of the particular feature exists, $P(features|label)$ represents the particular feature belongs to a label. In the real world, a data object may have thousands of words. For sentiment classification, most of them are useless, and they influence the results. Thus, Naı̈ve Bayes classifier is efficient and effective to build the model, to filter useless data and to improve the accuracy of data (Ting et al., 2011).

Critique

In sentiment classification, probabilistic classifiers rely heavily on feature selection because only high-quality features make probabilistic screening meaningful.

Relevance

This technique shows that Naı̈ve Bayes approach is good as a sentiment classifier based on appropriate feature selection. However, the author does not use this approach because feature selection in this research cannot become very precise. The

reason is given in section 2.2 Feature Selection. This is related to the establishment of the research objective and the research question.

2.3.6 Bayesian Networks Classifier

The Bayesian Networks model has an assumption that all the features are dependent. Comparing with the Naı̈ve Bayes classifier, Bayesian Networks try to go to another direction which is complicated to apply in research. If trying to get the graph that can describe Bayesian Networks, it contains nodes for variables and edges for conditional dependencies. The joint probability distribution of Bayesian Networks is expensive to use, so it is not very popular in machine learning.

When using the Bayesian Networks classifier for sentiment classification, the relation between features can be easily found because this is what Bayesian Networks focus on. Bayesian Networks can be expanded to more dimensions to combine more related variables as vectors. The semi-supervised framework could also be introduced to sentiment classification when adapting Bayesian Networks to deal with a large amount of unlabelled data (Huang and Bian, 2009).

Critique

In sentiment classification, features cannot be dependent or independent. Thus, Bayesian Networks have limitations in application.

Relevance

This technique shows that the Bayesian Networks approach is good as a sentiment classifier when dealing with related features. However, the author does not use this approach because the choice of data should not be so limited in this research. This is related to the establishment of the research objective and the research question.

2.3.7 Maximum Entropy Classifier

The maximum entropy classifier focuses on transforming labelled features to vectors which can be used to determine the weights of features. After analysing the weights, researchers can get the label of the feature with the highest probability. The probability of each label can be calculated by the following formula. $dotprod(weights, encode(fs, label))$ represents vectors of labelled features after encoding.

$$P(fs|label) = \frac{dotprod(weights, encode(fs, label))}{sum(dotprod(weights, encode(fs, l))forlinlabels)}$$

For sentiment classification, choosing distribution is the main job of maximum entropy classifier. Parallel sentences are normally considered as data objects. When a dataset is small, there are few distribution approaches because of the lack of training data. Thus, the maximum entropy classifier can be helpful because it considers weights and probability and the performance is better than common classifiers such as Naïve Bayes classifier (Ziebart et al., 2008).

Critique

A maximum entropy classifier is used widely when dealing with limited data. However, it may get problems when facing big data.

Relevance

This technique shows that the Maximum entropy approach is very effective under certain conditions. However, the author does not use this approach because datasets are not selected based on classifiers. This is related to the establishment of the research objective and the research question.

2.3.8 Unsupervised Learning

Unsupervised learning is a special approach to machine learning. If supervised learning is about teaching computers to do predictions based on built-in algorithms, unsupervised learning can be explained as doing predictions by the computer itself without a teacher. Computers no longer have 'correct answers' to deal with the mistakes they made in the training section. The aim of classification is dividing training data objects into different categories. Supervised learning relies on a big number of labelled features in data objects for training. However, when collecting data objects in the real world, they cannot be neat and organized. The ratio of omissions and errors is hard to control in a small range. Moreover, labelling features takes a lot of time as well. Thus, unsupervised learning is trying to handle these problems. Related research is still in the exploration stage (Hastie et al., 2009).

Critique

In sentiment classification, completely unsupervised learning is hard to use because of disadvantages in performance. Thus, some researchers use weak or semi-

supervised learning for exploring. Weakly supervised learning can be implemented at the level of features other than the level of instances. Sentiment lexicon would be used to label features. Then, results can be applied to a sentiment classifier model to recognize words with polarity. It performs well when dealing with texts in blogs or comments from IMDB and amazon.com (Medhat et al., 2014).

Relevance

This technique shows that an unsupervised learning approach may not be suitable for this research because its performance cannot be accurately measured. This is related to the establishment of the research objective and the research question.

2.3.9 Combined Classifier

When doing sentiment analysis, most researchers may not only pay attention to particular classification approaches. The first reason is although machine learning has developed so many years, some of the main algorithms have not yet been finalized. Researchers are still working hard to improve their performance. Thus, no one can know the current update status of all algorithms, which can be done by researchers is trying. Second, research topics are always closely related to reality. Therefore, researchers should try some different algorithms to identify whether they are suitable for the data model of the research topic. For example, topics can be recognized by researchers in online articles. The problems of data objects are that the distribution of different sentiment is not in balance, articles can change based on the market, models are not robust enough. In this case, a multi-algorithm should be introduced to the research such as Decision Tree, SVMs, Rule-based, Naive Bayes, and Bayesian Networks. Furthermore, some researchers think supervised learning and unsupervised learning can be combined to use at different levels of the research. In some cases, this new approach performs well (Medhat et al., 2014).

Critique

In sentiment classification, the combined classifier may have the flexibility to integrate the advantages of multiple approaches. However, it may cause unexpected problems at the same time. Researchers also need to avoid making it too complicated and losing the opportunity for further development.

Relevance

This technique shows the advantages of a combined approach. However, the purpose of this research is comparing different algorithms but not develop new approaches. Thus, the combined approach is not used in experiments. This is related to the establishment of the research objective and the research question.

2.3.10 Dictionary-based Approach

As mentioned in section 2.2 Feature Selection, linguistic experts also contribute a lot to sentiment analysis. In sentiment classification, there are also some lexicon-based approaches that are effective. Some researchers think that words with opinions are key elements of sentiment classification including some phrases and idioms. Thus, they collect them to make a dictionary with two main automatic approaches.

The dictionary-based approach has the following steps. First, some opinion words should be collected by researchers themselves because these words from the original data object in the research. Second, researchers can look for words related to original opinion words such as their synonyms and antonyms from dictionaries. As mentioned in the above sections, ‘seed words’ are the most important things in sentiment classification. Thus, in the next step, researchers should put newly discovered words into the list for ‘seed words’ and begins an iteration. The iteration stops when no new words exist. Finally, mistakes should be checked, and they should be deleted by researchers (Mohammad et al., 2009).

Critique

The disadvantages of the dictionary-based approach are the following. First, opinion words in context cannot be found. Second, keywords that may influence the meaning of whole sentences or paragraphs do not account for a large percentage of the ‘seed words’ list. These problems make the performance of classifiers do not meet the requirement of researchers. Furthermore, it takes a lot of time because of the large amount of manual work (Mohammad et al., 2009).

Relevance

This technique shows that a dictionary-based approach has the potential to develop. However, this approach involves an advanced design process that takes a lot of time, which leads to unfair comparisons. Thus, the dictionary-based approach is not used in experiments. This is related to the establishment of the research objective and the research question.

2.3.11 Corpus-based Approach

Corpus-based approaches focus on another sequence of steps to meet the requirement of sentiment classification. First, it looks for opinion words not only in words related to ‘seed words’ but also in the context. Thus, the approach needs to recognize syntactic patterns and use a large corpus. Second, most researchers start from adjective words because they always have strong sentiment. Third, researchers use constraints for seeking more ‘seed words’, such as finding ‘and’, ‘or’, ‘but’ in sentences or paragraphs. The meaning of this step is conjoining adjective words and related words because conjunction words always indicate a clear relationship between original words and expanded words. These steps cycle, then, a graph of sentiment words is formed, and all the words are divided into positive and negative.

The approach can be developed into other versions. For example, some researchers detect sentiment words by sequence learning techniques. They use the complicated dependencies between entities in the real world and words to build a relationship model. At the same time, a model based on interdependencies among relations is built, too. Thus, words and related entities have a strong connection which is hard to break. Researchers can visualize all the relationships and compare them on the computer, which is convenient to get results. Some researchers make a further improvement that they build a set of resources to identify words that are not related to the area of the resources. In this way, the accuracy of classification gets higher and the performance becomes better than before (Mohammad et al., 2009).

Critique

Normally, a corpus-based approach does not only rely on corpus because no corpus can cover all the English words. They are smaller than dictionaries which leads to a lack of information. However, the particular corpus is suitable for particular ‘seed words’, which helps determine the directions of searching expanding sentiment words (Mohammad et al., 2009).

Relevance

This technique shows that the corpus-based approach has the potential to develop. However, this approach involves a special feature selection process, which leads to unfair comparisons. Thus, the corpus-based approach is not used in

experiments. This is related to the establishment of the research objective and the research question.

2.3.12 Other Approaches

Some approaches cannot be identified as a machine learning approach. However, they do not belong to the lexicon-based approach either. Formal Concept Analysis (FCA) is this kind of approach which is based on a mathematical approach. FCA develops a basic structure called 'formal concept' which is built by recognizing particular data objects and their features. Then, 'formal concept' can identify relationships between data objects and visualize them. The advantages of this approach are reducing the inherent ambiguities, decreasing the noise and dealing with uncertainty (Priss, 2006).

Critique

The disadvantage of this approach is that it cannot be easily applied in codes.

Relevance

The author runs different algorithms in the same computing environment to compare the pros and cons. Thus, the mathematical approach is not used in experiments because it cannot be encoded easily. This is related to the establishment of the research objective and the research question.

2.3.13 Summary of Sentiment Classification

In section 2.3 Sentiment Classification, the author introduces many different classifiers. In machine learning approach, there are decision tree classifier, support vector machines classifier, neural network classifier, rule-based classifier, Naïve Bayes classifier, Bayesian Networks classifier, maximum entropy classifier, unsupervised learning classifier, and combined classifier. In the lexicon-based approach, there are dictionary-based classifier and corpus-based classifiers. Moreover, there is a Formal Concept Analysis classifier which is neither machine learning approach nor lexicon-based approach. Each classifier has advantages and disadvantages. However, they are all heavily used in the study of sentiment analysis.

2.4 DEEP NEURAL NETWORK APPROACH

The general structure of a standard neural network (NN) is demonstrated in section 2.3.3 Neural Network Classifier, deep neural networks developed from basic neural networks appeared many decades ago. In the early stage of sentiment analysis, deep neural network approach was seldom applied because it took up too much computing power on computers. However, with the development of computer hardware like the central processing unit (CPU) and graphics processing unit (GPU), the computing power of computers is getting stronger, so that deep neural network approach which was not efficient performs much better than before. Since 2000, pure supervised deep learning (DL) with neural networks have achieved huge improvement. For example, a convolution neural network (CNN) is suitable for simulating the function of the visual cortex when processing image information. It has been silent for many years and has ushered in explosive growth in recent years. In 2012, CNN defeated many well-known traditional machine learning approaches and significantly improved the performance in the MNIST database (LeCun et al., 2015). Thus, researchers who are interested in sentiment analysis start to turn to deep neural networks for their projects (Schmidhuber, 2015). The author covers three deep neural networks that can be used in the field of sentiment analysis as shown in Figure 2-5.

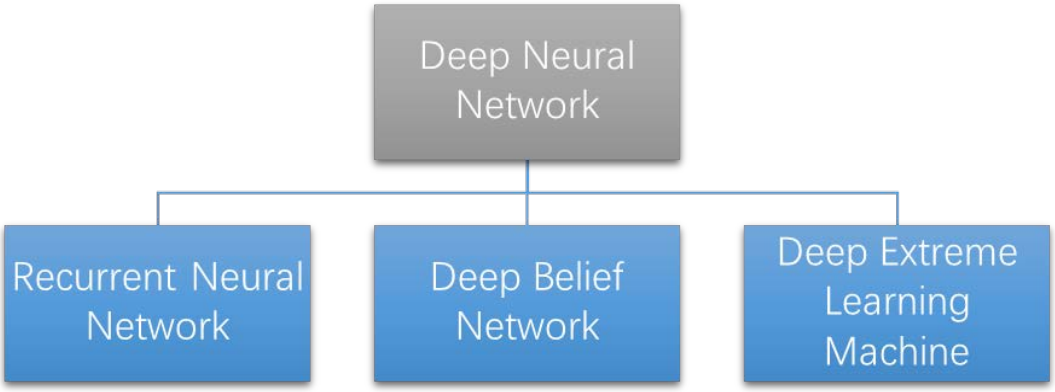

Figure 2-5 Deep Neural Network Approach

2.4.1 Recurrent Neural Network

Recurrent neural network (RNN) is suitable for research in NLP because of its special characteristic which is helpful when dealing with data based on text. For

example, RNN can learn objects, not in traditional sequences and rules. It can process parallel information and apply their mixed sequential program, which declines computation cost significantly (Sainath et al., 2015).

Normally, the process of sentiment classification using RNN can be explained in the following Figure 2-6. The first step is using vectors to represent words. In the second step, RNN is introduced to combine a list of words in one document based on semantic principles (Socher et al., 2013). In the third step, all documents have their matrixes. These are considered as final inputs of RNN. In the fourth step, the cost function of RNN should be calculated in every epoch in the training process. In the fifth step, parameters in RNN should be updated by back-propagation, and results are generated by the whole system (LeCun et al., 2015).

Figure 2-6 RNN for Sentiment Analysis (LeCun et al., 2015)

RNN condition language models by using special architecture. A typical RNN can be introduced in Figure 2-7 in detail. $x_{(1)}$, $x_{(2)}$, $x_{(3)}$ and $x_{(4)}$ are single words as inputs. Here is just an example, and there should be many more words in real NLP tasks. *E* represents the process of the encoder which means turning words into vectors. $e_{(1)}$, $e_{(2)}$, $e_{(3)}$ and $e_{(4)}$ are single word vectors as real inputs for RNN models. $h_{(0)}$, $h_{(1)}$, $h_{(2)}$, $h_{(3)}$ and $h_{(4)}$ are the results of computing the outputs of the hidden layer at every time-step t. $W_{(e)}$ and $W_{(h)}$ are basic parameters in RNN models, and they never change with the time-step. *U* represents the parameter for the final output of the model. The following equations can explain the relationship of variables in Figure 2-7 (LeCun, Bengio & Hinton 2015, p. 436).

$$h_{(t)} = \sigma(W_{(h)} * h_{(t-1)} + W_{(e)} * e_{(t)})$$

$$y_{(t)} = softmax(U * h_{(t)})$$

In the first equation, σ represents a non-linear operation, such as $\tanh()$. In every time-step, two inputs are given to the hidden layer. Word vectors $e_{(t)}$ are multiplied by the weight matrix $W_{(e)}$ and previous layers' outputs $h_{(t-1)}$ are

multiplied by the weight matrix $W_{(h)}$, and their sum is considered as a variable of the σ function. $h_{(t)}$ is the final output of this layer and this time-step. The connection of $h_{(t-1)}$ and $h_{(t)}$ makes RNN have 'memory', which can detect the information embedded in sequences. Keeping $W_{(e)}$ and $W_{(h)}$ the same value in different time-step leads to a small number of parameters in the RNN, which solves the dimensionality problem.

In the second equation, $softmax()$ calculates the probability distribution in every time-step. $y_{(t)}$ represents the prediction based on all the given context and the word vector in the current time-step. This structure is commonly used in classification tasks.

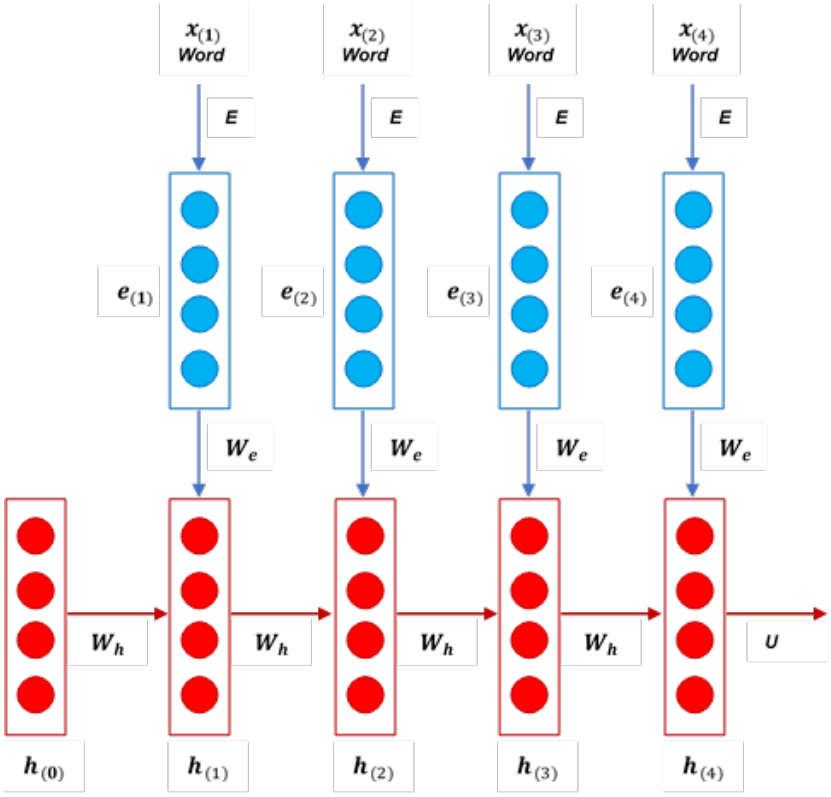

Figure 2-7 RNN Language Model (LeCun et al., 2015)

RNNs can also be considered as a loop if researchers simplify it. It can deal with sequence data with any length. However, when the lengths of sequences increase, the model sizes would not increase with them because the number of parameters is related to the number of recurrent layers and sizes of word vectors. Only the number of loops would increase (Maknickiene et al., 2018).

Critique

When the corpus is big, the training time of RNN would be long because it should calculate equations in every step following the order. After many steps, there may be vanishing gradients and exploding gradients in training (Zhang et al., 2016b).

Currently, in real NLP tasks, long short-term memory (LSTM) which is an improved version of RNN performs better in conducting the meaning of longer expressions like a long sentence or a whole paragraph. Sentence composition is one of the outstanding features of LSTM. Words in LSTM are low dimensional vectors with real value (Varior et al. 2016). One matrix represents one sentence or one paragraph, and it contains many word vectors. In detail, sentences with different lengths are represented by fix-length vectors, which can show the sequences of words in sentences. In the level of documents, RNN can also represent sentences by fix-length vectors. However, standard RNN always meets the problem of gradient vanishing or exploding, which makes long-distance hard to model. Thus, the cooperation between LSTM and gated neural network solves the problem. It can adaptively select input vectors and remove history vectors based on semantic composition, and the last hidden vector is considered a result of sentiment classification (Tang et al., 2015a).

LSTM condition language models by using the special architecture called gates. A typical LSTM can be introduced by Figure in detail. $i_{(t)}$ represents the input gate, $f_{(t)}$ represents the forget gate, $o_{(t)}$ represents the output gate, $c_{(t)}$ represents the final memory cell and $h_{(t)}$ represents the final output. The following equations can explain the relationship of variables in Figure (Goodfellow, Bengio & Courville 2016).

$$i_{(t)} = \sigma(W_{(i)} * x_{(t)} + U_{(i)} * h_{(t-1)})$$

$$f_{(t)} = \sigma(W_{(f)} * x_{(t)} + U_{(f)} * h_{(t-1)})$$

$$o_{(t)} = \sigma(W_{(o)} * x_{(t)} + U_{(o)} * h_{(t-1)})$$

$$s_{(t)} = \sigma(W_{(c)} * x_{(t)} + U_{(c)} * h_{(t-1)})$$

$$c_{(t)} = f_{(t)} \circ c_{(t-1)} + i_{(t)} \circ s_{(t)}$$

$$h_{(t)} = o_{(t)} \circ tanh(c_{(t)})$$

In the first equation, σ represents a non-linear operation, such as $\tanh()$. In every time-step, the process in the input gate is basically the same as that in traditional RNN. Word vectors $x_{(t)}$ are multiplied by the weight matrix $W_{(i)}$ and

previous layers' outputs $h_{(t-1)}$ are multiplied by the weight matrix $U_{(i)}$, and their sum is considered as a variable of the σ function. $i_{(t)}$ makes RNN have 'memory'. However, there is another calculation step shown in the fourth equation. Word vectors $x_{(t)}$ are multiplied by the weight matrix $W_{(c)}$ and previous layers' outputs $h_{(t-1)}$ are multiplied by the weight matrix $U_{(c)}$, and their sum is considered as a variable of the σ function. $s_{(t)}$ makes RNN have 'new memory' (Ma et al. 2015).

Calculation about the forget gate is shown in the second equation, σ represents a non-linear operation, such as $\tanh()$. Word vectors $x_{(t)}$ are multiplied by the weight matrix $W_{(f)}$ and previous layers' outputs $h_{(t-1)}$ are multiplied by the weight matrix $U_{(f)}$, and their sum is considered as a variable of the σ function. $f_{(t)}$ makes RNN has 'forget' function. In the fifth equation, $\circ$ represents element-wise multiplication, and final memory is generated by both 'new memory' and 'forget'.

Calculation about output gate is shown in the third equation, σ represents a non-linear operation, such as $\tanh()$. Word vectors $x_{(t)}$ are multiplied by the weight matrix $W_{(o)}$ and previous layers' outputs $h_{(t-1)}$ are multiplied by the weight matrix $U_{(o)}$, and their sum is considered as a variable of the σ function. $o_{(t)}$ breaks the connection between final memory and hidden state. In the sixth equation, $\circ$ represents element-wise multiplication, and information that does not need to exist in the hidden state is filtered out.

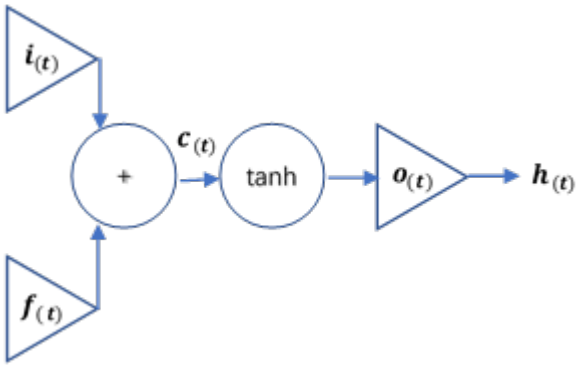

Figure 2-8 Internal Structure of LSTM (Goodfellow et al., 2016)

LSTM achieved state-of-the-art performance in many NLP tasks. By implementing word embedding techniques such as GloVe developed by Stanford, LSTM and its deformation can be competitive for classification tasks (Wang et al., 2016b).

Critique

However, LSTM's text processing capabilities can be further improved by applying many new architectures, such as Bidirectional LSTM and attention-based LSTM. Furthermore, when applying in small datasets, overfitting often occurs during the training of LSTM. Thus, approaches to mitigate overfitting need to be tried, such as dropout and normalization.

Relevance

This technique shows that the LSTM approach is suitable for comparison because it achieves the best performance in sentiment classification tasks. The author can detect the advantages and disadvantages of DELM clearly by applying LSTM and DELM in the same environment and datasets. This is related to the establishment of the research objective and the research question.

2.4.2 Deep Belief Network

Human language is complex and hard to classify in a clear principle. Deep neural networks provide a good chance for developing a more accurate sentiment classifier with the help of millions of parameters. However, when training models, the performance of algorithms would not be that good because of the parameter tuning process. Moreover, supervised training needs a large number of training data that requires manual labelling. Many popular deep neural networks including CNN and RNN the author mentioned in the above sections rely on this.

There are also some solutions to this problem, one of them is a semi-supervised learning approach. The author has briefly described its use in section 2.3.8 Unsupervised Learning. Similarly, this approach can also be applied to deep neural networks. Deep belief network (DBN) with special hidden layers use a greedy learning algorithm to find good parameters fast (Wang et al., 2016a). It is an unsupervised learning approach but can be used as a classification approach for labeled data (Hinton et al., 2006). Thus, some researchers have a strong interest in applying it to sentiment classification.

When building the structure of DBN, there are two steps. In step one, a restricted Boltzmann machine (RBM) is introduced into the basic structure. This model consists of an input layer and a hidden layer that have symmetrically weighted connections between each other and no inner connection in themselves (Abdel-Zaher et al., 2016). The basic structure of RBM is shown in Figure 2-9. DBN is constructed by many RBMs to get better performance. Moreover, greedy layer-wise unsupervised learning should also be used. In DBN, the priority is not completing forward-propagation and backward-propagation as popular CNN and RNN. Alternatively, it assumes higher layers do not exist when training lower layers. Thus, this densely connected belief network train layer-by-layer (Dedinec et al., 2016). In step two, supervised learning plays an important role. DBN applies an inefficient fine-tuning algorithm to adjust parameters in the model based on labels. The unsupervised learning approach in step one is used for initializing the weights in supervised learning (Hinton et al., 2006).

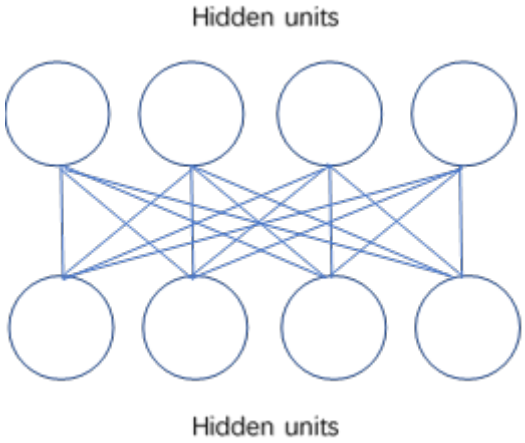

Figure 2-9 Restricted Boltzmann Machines in DBN (Hinton et al., 2006)

RBM can be described by the following equations.

$$E(\boldsymbol{v}, \boldsymbol{h}) = -\sum_{i=1}^{n}\sum_{j=1}^{m} w_{ij} h_i v_j - \sum_{j=1}^{m} b_j v_j - \sum_{i=1}^{n} c_i h_i$$

$$p(\boldsymbol{h}|\boldsymbol{v}) = \prod_{i=1}^{n} p(h_i|\boldsymbol{v})$$

$$p(\boldsymbol{v}|\boldsymbol{h}) = \prod_{j=1}^{m} p(v_j|\boldsymbol{h})$$

In the first equation, $E(\boldsymbol{v}, \boldsymbol{h})$ represents the energy of an RBM block. $\boldsymbol{v}$ can be considered as m random variables in the visible layer. $\boldsymbol{h}$ represents n random variables in the hidden layer. w_{ij} is the weight with a real value between h_i and v_j, and b_j and c_i are bias terms. In the second equation and the third equation, since there is no connection between nodes in the same layer, visible variables are independent when a particular hidden variable is given and vice versa. When conducting DBN using RBM blocks, these two equations can be used to generate parameters in this feed-forward neural network with the greedy algorithm (Zhao et al., 2017).

DBN has many advantages as a feed-forward neural network. First, it can learn features without requiring back-propagation or other information from labels. Overfitting problems should be solved easily because of this. Second, the results it learned can be checked easily. Third, representations in deep hidden layers can be extracted. Fourth, classification tasks may achieve good performance by using DBN (Hinton, Osindero & Teh 2006, p. 1547-1548).

Critique

However, DBN's text processing capabilities are questionable because it can only add more hidden layers to fit the data better but cannot solve the dimensionality problem effectively. If the corpus is big, using DBN would be computationally expensive.

Relevance

This technique shows that the DBN approach is suitable for comparison because it is a classic deep neural network in sentiment classification tasks. The author can detect the advantages and disadvantages of DELM clearly by applying DBN and DELM in the same environment and datasets. This is related to the establishment of the research objective and the research question.

2.4.3 Deep Extreme Learning Machine

Deep extreme learning machine (DELM) is a novel deep neural network approach compared with CNN, RNN, and DBN. It is designed based on feedforward networks that use the white box for multiple hidden layers. As mentioned in the above sections, sentiment classification is a typical classification task in NLP. Thus, DELM may also show its good performance in sentiment analysis as other deep neural networks (Sun et al., 2017).

DELM is developed from an extreme learning machine (ELM) which is a good example of single hidden layer feedforward networks (SLFNs). In the beginning, ELM was designed for processing numbers (Zhang et al., 2016a). However, when applying on MNIST dataset, it showed advanced learning accuracy and training speed in image processing tasks. Thus, researchers turn to it for a fast solution. The theory can be explained by the following equations (Huang et al., 2006).

$$h_i(\boldsymbol{x}) = G_i(\boldsymbol{a}_i, b_i, \boldsymbol{x})$$

$$f_L(\boldsymbol{x}) = \sum_{i=1}^{L} \beta_i G(\boldsymbol{a}_i, b_i, \boldsymbol{x})$$

$$h(\boldsymbol{x}) = [G(\boldsymbol{a}_1, b_1, \boldsymbol{x}), \dots, G(\boldsymbol{a}_L, b_L, \boldsymbol{x})]$$

In every nonlinear piecewise continuous hidden node, the process of calculation can be explained in the first equation, $h_i(\boldsymbol{x})$ represents outputs of nodes. $G_i()$ means non-linear operations, such as sigmoid networks, RBF networks, threshold networks, high-order networks, wavelet networks, and polynomial networks. $\boldsymbol{a}_i$ and b_i are weights in weight matrix and bias, and the input x is considered as the variable of the $G_i()$ function. In generalized SLFNs, the output can be generated by the second equation. Suppose there are L random hidden neurons, the third equation can explain the output of the whole hidden layer including all the nodes. In ELM, hidden neurons would never be modified manually because parameters are assigned randomly in this neural network. Thus, β_i would be an important hyper-parameter in ELM for minimizing $f(\boldsymbol{x}) - f_L(\boldsymbol{x})$ (Gu et al., 2015).

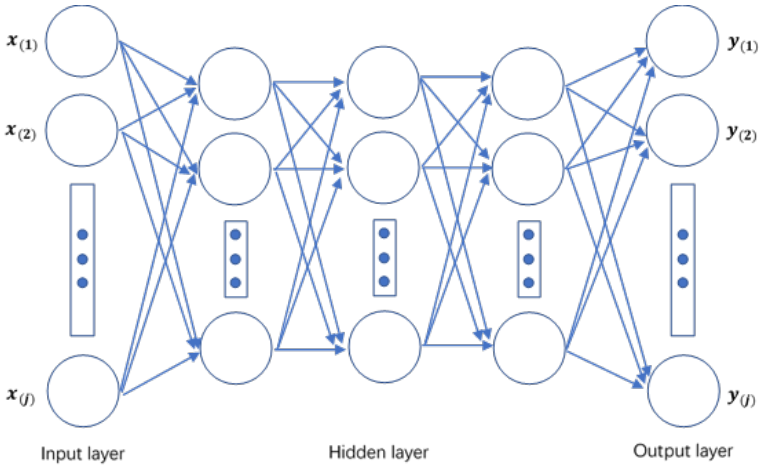

Figure 2-10 Structure of DELM (Tang et al., 2015b)

The structure of DELM is shown in Figure 2-10. By using the technique named auto-encoder, ELM can be transferred to DELM. The hidden layers would remain the same feature space as they were in single layer ELM. Parameters in DELM are also not required to be iteratively tuned as parameters in a single layer. Because of not relying on back-propagation, DELM has much fewer overfitting problems than other deep neural networks such as CNN and RNN. Furthermore, DELM achieved state-of-the-art results in image classification on the MNIST dataset. It spent the least training time as 281.37 seconds and got the highest accuracy as 99.13% ELM (Tang et al., 2015b).

DELM not only achieved good results in the field of picture recognition but also had many applications in the field of NLP in recent years. It could be used with many other algorithms. For example, in NLP tasks, it could be used as an embedding layer above a hidden layer in other deep neural networks and achieve a better result than traditional word-embedding layers. In sentiment analysis, with the help of statistical feature selection approaches, DELM performed better than support vector machine (Wang and Parth, 2016).

DELM is a tuning-free deep neural network. Its learning time is extremely short, and it can maintain high accuracy at the same time because of using matrix operation. In the structure, parameters in hidden layers are independent of the training. The parameters are generated before getting the outputs. DELM can be used in multiple tasks such as clustering, feature learning, classification, and regression (Yu et al., 2015).

Critique

However, DELM's text processing capabilities can be further improved by applying many new architectures. Although it can be used for word-embedding tasks, in classification tasks, word-embedding is hardly applied because of its structure. Furthermore, when applying in large datasets, dimensionality problem and sparsity problem should be solved in the future.

Relevance

This technique shows that the DELM approach is suitable for comparison because it has potential in classification tasks. The author can detect the advantages and disadvantages of DELM clearly by applying DELM, LSTM, and DBN in the same

environment and datasets. This is related to the establishment of the research objective and the research question.

2.4.4 Other Deep Neural Networks

So far, LSTM is still the most popular neural network approach for NLP including sentiment analysis. Researchers are still putting efforts into searching better approaches. There are two main directions for new research topics in sentiment analysis. One is improving existing algorithms to improve their performance such as accuracy, stability, and speed. Another one is developing new algorithms or applying algorithms from other fields for sentiment analysis.

Main fields for improvement are layers in LSTM and pre-processing section when conducting LSTM. As mentioned in the above sections, sentiment analysis can be used in many different areas. Thus, the dataset can be small or big, the data can be easy to label or hard to label, the distribution can be suitable for training or difficult for training. Researchers should find the relationship between which kind of pre-processing approach is suitable for a particular kind of dataset. Then, when doing classification, deep neural networks always have many layers to process the data which is represented by vectors and matrix. Sometimes, particular layers may have some significant effects on a particular matrix, which can improve the performance of the neural network. However, some deep neural networks have a black box structure which makes identifying the relationship between layers and performance hard. Researchers always change some parameters as input to check whether new layers are useful (Yamada and Kinoshita, 2018).

Novel algorithms like DELM can also have some particular characteristics which are helpful when classifying sentiment. However, researchers may take a lot of time to apply these approaches to research because of the lack of pre-study. Another popular way is combining different neural networks for sentiment classification. Some researchers use user memory network (UMN) and product memory network (PMN) with LSTM, which achieves a better performance in tweets analysis.

Critique

Unpopular deep neural networks always achieve much less attention than popular deep neural networks such as LSTM. Thus, the new architecture of these deep neural networks will not be developed. Besides, the latest feature selection approaches

would not be used in experiments together with these deep neural networks, and the most powerful computing power would not be used in research related to them.

Relevance

These techniques show that some deep neural networks are not suitable for this research because they are not fully developed and not reliable enough. This is related to the establishment of the research objective and the research question.

2.4.5 Summary of Deep Neural Network Approach

In section 2.4 Deep Neural Network Approach, the author introduces many different deep neural networks for sentiment classification. In the deep learning approach, there are CNN, RNN, DBN, and DELM. Moreover, there are many variants of mainstream deep neural networks and unpopular deep neural networks, but they are not heavily used in the study of sentiment analysis.

2.5 SUMMARY AND IMPLICATIONS

In conclusion, after doing literature reviews on sentiment analysis, the author finds that there are three typical domains in sentiment analysis. In the first domain, researchers investigate the background and identify objects of sentiment analysis. The second domain, feature selection or data pre-processing is applied by researchers to clean the raw data and to translate it into a data format that the computer can understand. The last domain, sentiment classification approaches train machine learning or deep learning models by input data and labels. Then, researchers use these models to predict new objects' sentiment. This is roughly the same as the sentiment analysis process that the author learned before doing the literature review. That is discussed in section 2.1.5 Process of Sentiment Analysis.

Furthermore, the author finds that both problems and opportunities exist in the sentiment analysis research field.

First, sentiment analysis can be applied in many jobs, but there are still many jobs that have no requirement in sentiment analysis. The reason would be that the requirement has not been discovered yet. Thus, the opportunity for researchers is discovering more jobs who need sentiment analysis and detecting target objects in these jobs. This can effectively promote the development of sentiment analysis research.

Second, traditional machine learning based approaches encounter bottlenecks in sentiment analysis. Although the accuracy and speed are growing by optimization, the performance cannot achieve commercial application level yet. Thus, more detailed questions like ‘which feature selection approach is necessary’ need to be solved by researchers to develop some new approaches which are more suitable for sentiment analysis.

Third, deep neural networks develop so fast and perform well recently because they can get help from millions of parameters, and the computing power is provided by better computers. In sentiment analysis, deep neural networks have shown many advantages compared with traditional machine learning classifiers. Researchers who are interested in sentiment analysis should pay more attention to deep neural networks. Of course, there are still some difficult problems in this area such as ‘how to choose better feature selection approaches for deep leaning’ and ‘how to improve the generation ability of deep learning models’. However, if the problems were solved one by one, sentiment analysis research will reach a new level with the help of deep neural networks. The author believes that the accumulation of details leads to an overall breakthrough.

In the next chapter, the author designs a series of approaches to implement a complete sentiment analysis process.

Chapter 3: Methodology

This chapter describes the design adopted by this research to achieve the research objective stated in section 1.3 of Chapter 1. Section 3.1 discusses the research philosophy used in the study; section 3.2 details the research strategy in the study; section 3.3 lists the whole picture of research design in the study including the main approach and the procedure of experiments. Justification of four distinct phases' use is covered in the following four sections; section 3.4 outlines the reasons for datasets selection; section 3.5 discusses how the data was pre-processed; section 3.6 demonstrates how the deep neural networks were applied for classification tasks; section 3.7 indicates how the performance of experiments was evaluated; finally, section 3.8 makes the conclusion of the methodology.

3.1 RESEARCH PHILOSOPHY

Sentiment analysis can be considered as the ontology of this research. In daily life, sentiment can be found everywhere in society because this is related to the basic characteristic of people. Commonly, the sentiment is subjective and hard to value since this is a linguistic definition that belongs to social science. For example, babies may think that there are only two types of sentiment – happy and unhappy, young people may consider multiple sentiments such as affection, enjoyment, amusement, contentment, grief, loathing, uneasiness, and shame (Serrano-Guerrero et al., 2015). However, in sentiment analysis which is a field of study combining linguistics and computer science, sentiment can be considered as an objective thing that can be ranked.

The type of epistemology in sentiment analysis is positivist because researchers have a set of established research approaches and processes in this field of study. In another word, there are some laws of cause and effect in the logical chain of sentiment analysis. The objectivity of sentiment analysis can be achieved by a rigorous research paradigm design which is covered in section 3.3 Research Design.

3.2 RESEARCH STRATEGY

Currently, DELM has shown its good performance in many other fields of study such as text mining and stock prediction (Li et al., 2016). The relevant technical details have been discussed in the previous chapter. However, the research on the application of DELM in the field of sentiment analysis lacks enough contrast with mainstream deep neural networks. Moreover, there are not enough datasets suitable for deep learning as the original data source was applied in the previous related research.

Therefore, the purpose of this research is to detect DELM's ability to perform sentiment analysis tasks. The author conducts a series of experiments to apply multiple deep neural networks to analyse the sentiment embedded in the data in multiple datasets for this purpose.

The purpose of the research can also be considered as testing the performance of DELM in sentiment analysis including its strengths and weakness. The test can be achieved by conducting a particular sentiment analysis project in DELM following a common research paradigm in this field of study. All the related data can be collected by the author and it can be divided into different categories and filled in multiple tables

for further analysis. Of course, there are at least two other groups of controlled experiments for this research. In the original design, sentiment analysis using LSTM and DBN which is mentioned in section 2.4 Deep Neural Network Approach is also implemented following the same research paradigm in the DELM project. According to quantitative analysis and some statistical experiments, if the performance of DELM can reach a significant level, the report based on data about the performance of three different deep neural network approaches can demonstrate that DELM has a better performance in sentiment analysis.

The primary research objective for this thesis is the following:

- To establish whether DELM achieves better accuracy and takes less training time when conducting sentiment classification tasks on six text-based data sets in different domains, in comparison with LSTM and DBN.

In addition to the above objective, there are the following sub-objectives:

- To identify and select suitable text-based datasets in different domains for sentiment classification tasks based on deep neural networks.
- To identify and select the most suitable feature selection approach for sentiment classification tasks based on deep neural networks.
- To evaluate the quality of three deep neural networks used to classify text-based data.

The primary research question for this thesis is "Does DELM have advantages on accuracy and training time over other deep neural network algorithms in sentiment analysis?"

3.3 RESEARCH DESIGN

This research uses a quantitative analysis approach to solve all the research questions. In quantitative analysis, there are a large number of research approaches such as data collection by experiment, causal comparatives, and hypothesis formulation. Moreover, statistical techniques such as descriptive statistics, confidence intervals, and data visualization are also used after getting the results. In the results analysis step, normality, parametric testing, and non-parametric testing are generated by the author, too (Ghasemi and Zahediasl, 2012).

Quantitative analysis is suitable for this research. The reasons are sufficient as following: First, the research objectives mentioned in section 3.2 Research Strategy mean that this research collects numerical data, such as accuracy and training time. Quantitative analysis is designed for dealing with numerical data. Second, the objective, positivist research philosophy always falls in quantitative analysis. As mentioned in section 3.1 Research Philosophy, the ontology of this research is objective, and the epistemology of this research is positivist. Third, precisely measuring variables and testing hypotheses is essential in this research, which is a significant part of quantitative analysis (Bernard, 2013). Forth, a large sample size is applied in this research. Quantitative analysis is good for improving the reliability of dealing with a big amount of data in the research. Fifth, quantitative analysis reduces bias to make results more convincing. Sixth, quantitative analysis costs less money and time because the author does not need to design a survey or organize some interviews (Babbie, 1998).

The steps for sentiment analysis in this research can be considered as four distinct phases. These phases are generally demonstrated here and more in-depth in the next four sections: (1) Datasets Selection, (2) Text-based Data Pre-processing, (3) Deep Neural Network Classification and (4) Performance Evaluation. The data used in this research is from the well-known data science website -- Kaggle. In this research, six datasets from Kaggle were used in experiments. The rules and procedures for selecting appropriate datasets are given in the next section 3.4 Datasets Selection. The second phase converts texts to ‘tensors’ using pre-processing approaches that are commonly used in NLP tasks and details is be covered in section 3.5 Text-based Data Pre-processing. In the third phase, three deep neural network approaches are applied to classify the distilled data, which is shown in section 3.6 Deep Neural Network Classification. In the last phase, several statistical tools are used to analyse the difference in accuracy and training time for different data sets and different depth neural network approaches. Which tools are used and why they are used are answered in section 3.7 Performance Evaluation. The flowchart of the process of sentiment analysis in this research can be seen in Figure 3-1.

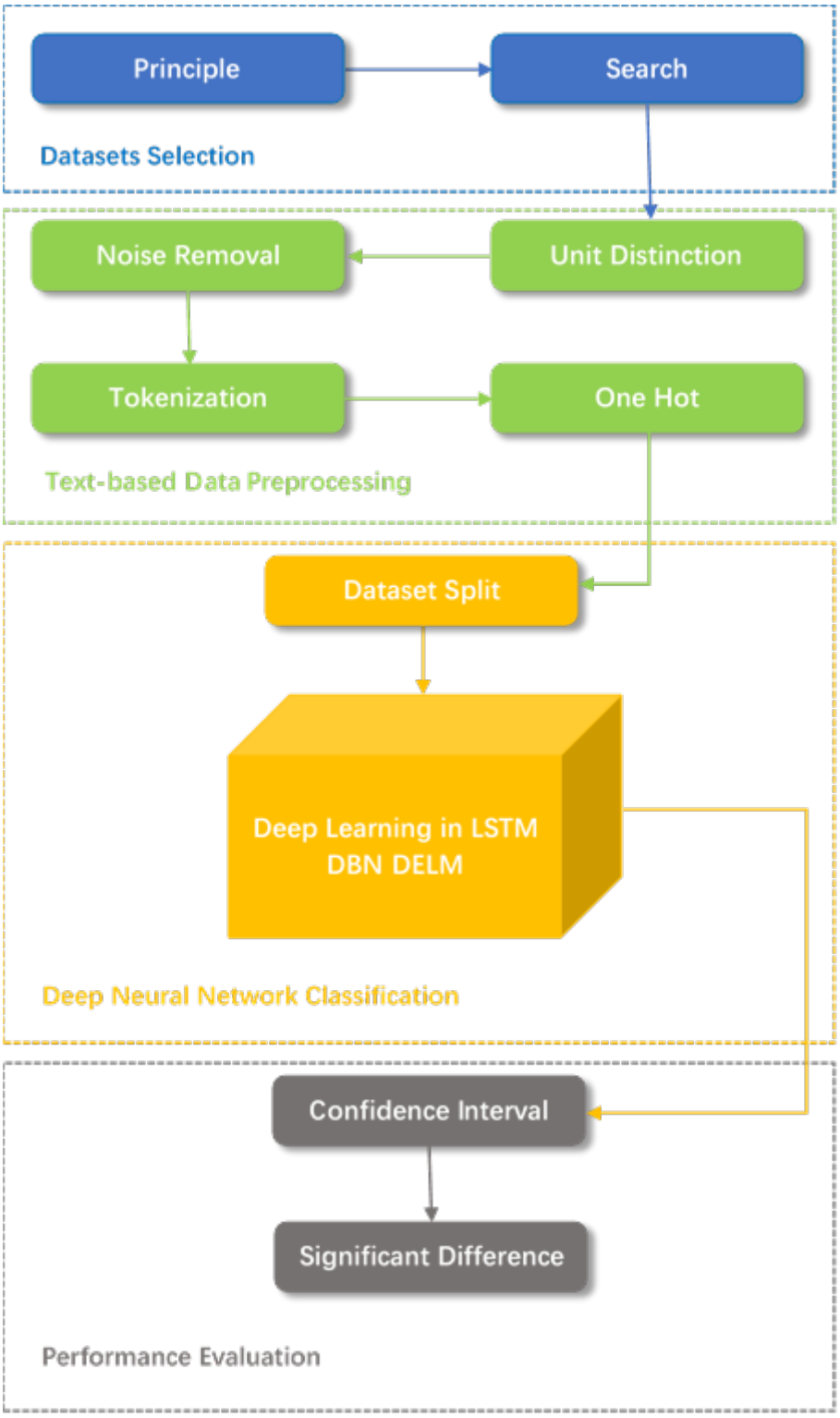

Figure 3-1 Phases of Sentiment Analysis

3.4 DATASETS SELECTION

This section discusses the principles for selecting the most suitable datasets for sentiment analysis in this research. First, when classifying sentiment, it is necessary to

have specific data carriers containing sentiment information. As mentioned in the literature review chapter section 2.1.3 Applications, these carriers can be videos, audios, texts or mixed stuff. However, currently, video-based datasets and audio-based datasets are not fully developed, and they are not frequently used in sentiment analysis. Existing video-based datasets and audio-based datasets have limited stability and a lot of noise. Furthermore, these kinds of data are hard to process before classifying by deep learning algorithms. Thus, this research uses only text-based data for sentiment classification. Second, this research conducts supervised learning for classification because the results of it are easier to evaluate than unsupervised learning. When implementing supervised learning, labels play the role that cannot be lost. Thus, datasets in this research should already have clear labels since this research does not have experts to label the data. Third, the author uses three deep neural networks as classifiers in this research and wants to compare their performance. Due to that deep neural networks are powerful in their feature extraction capabilities, comparing and highlighting this capability is an important part of comparing the performance. Therefore, the content of the datasets should be widely distributed in different topics. Fourth, sparsity is a big problem in NLP tasks. The ability to deal with high sparse data is one of the important principles for determining whether a deep neural network is suitable for NLP tasks. Accordingly, selected datasets should be in different sparsity levels because this makes the results of the performance of different deep neural networks more convincing.

After searching datasets online using the principles mentioned above and using ‘sentiment analysis’ as keywords, the author finds six datasets for this research. The first one is the IMDB review dataset (Table 3-1). It contains enough text-based data with labels. The original labels refer to scores in the range of 10. However, to simplify the use, researchers made ‘negative’ labels for reviews have scored less than 5 and made ‘positive’ labels for reviews have scored more than 6. Labels ‘neutral’ are not included in this dataset (Pang and Lee, 2004).

Table 3-1 Examples of Labels and Data in The IMDB Review Dataset (Pang and Lee, 2004)

Label	Data
Positive	first think another Disney movie, might good, it's kids …

Negative	Put aside Dr. House repeat missed, Desperate ...
…	…

The second one is the Amazon review dataset (Table 3-2). It consists of Amazon customer reviews for input text and star rating for output labels. However, star rating information is transferred to binary sentiment tags as ‘positive’ and ‘negative’. The language of the reviews in the dataset is rich, which results in a very high level of sparsity when applying language models (Pang and Lee, 2008).

Table 3-2 Examples of Labels and Data in The Amazon Review Dataset (Pang and Lee, 2008)

Label	Data
Positive	I hope a lot of people hear this cd. We need more …
Negative	This is a self-published book, and if you want to know …
…	…

The third one is the Hotel review dataset (Table 3-3). The data in it was scraped from Booking.com which is a famous website containing travel information. The original data includes 17 fields such as the address of hotels, date of reviews, names of hotels, negative reviews and positive reviews. However, the author only intercepts two columns called ‘negative reviews’ and ‘positive reviews’ for sentiment analysis (De Albornoz et al., 2011).

Table 3-3 Examples of Labels and Data in The Hotel Review Dataset (De Albornoz et al., 2011)

Label	Data
Positive	Location was good and staff were ok It is cute hotel …
Negative	I am so angry that I made this post available via all …
…	…

The fourth one is the US airline sentiment dataset (Table 3-4). It originally came from ‘Crowdflower's Data for Everyone’ library. All the tweets in this dataset are related to six US airlines. The language of the content in the dataset is limited in the airline domain, which results in a relatively low level of sparsity when applying language models. Labels in this dataset have three types -- ‘positive’, ‘negative’ and ‘neutral’ (Wan and Gao, 2015).

Table 3-4 Examples of Labels and Data in The US Airline Sentiment Dataset (Wan and Gao, 2015)

Label	Data
Positive	@VirginAmerica plus you've added commercials to the …
Negative	@VirginAmerica it's really aggressive to blast obnoxious …
Neutral	@VirginAmerica What @dhepburn said.
…	…

The fifth one is the Twitter dataset (Table 3-5). It consists of millions of tweets with labels (Pak and Paroubek, 2010). The length of content in every unit (tweets) is relatively short when compared with the first dataset and the second dataset. It is the most frequently used dataset in NLP tasks (Bao et al., 2014).

Table 3-5 Examples of Labels and Data in The Twitter Dataset (Bao et al., 2014)

Label	Data
Positive	I handed in my uniform today. I miss you already
Negative	It is so sad for my APL friend.....
…	…

The last one is sarcasm on the Reddit dataset (Table 3-6). Millions of sarcastic comments are in this dataset. In particular, the data in this data set is not labelled as

simple ‘positive’ and ‘negative’. The labels reflect that the comment is in jest or not meant to be taken seriously (Thelwall and Buckley, 2013).

Table 3-6 Examples of Labels and Data in The Reddit Dataset (Thelwall and Buckley, 2013)

Label	Data
Positive	wow it is totally unreasonable to assume that the agency ...
Negative	You do know west teams play against west teams more …
…	…

3.5 TEXT-BASED DATA PRE-PROCESSING

This section demonstrates the approaches used to pre-process raw data in selected datasets. Text-based data pre-processing is essential because classifiers based on machine learning or deep learning cannot take texts as input directly. Besides, there are too many noises in raw data which would disturb the feature extraction process of classifiers. Hence, classifiers would learn incorrect features, and this would affect the accuracy of the classification.

Figure 3-2 Steps in Text-based Data Pre-processing

The phase of data pre-processing has four basic steps. These steps are shown in the figure above. First, a unit distinction should be applied in this phase. As we knew,

sentiment information is embedded in units such as sentences, paragraphs or documents. A single word may have a strong sentiment tendency, but its meaning would be significantly different in different units. In this research, the author would classify the sentiment of sequences of words like most research in sentiment analysis (Godbole et al., 2007). Thus, the author separates the units in raw data at the beginning of further processing.

Second, noise removal is also a crucial step, and it can be called stemming. In online texts, there are a lot of noises. These noises can be divided into the following types. First, people are more relaxed when they are on the Internet so that they tend to ignore grammar rules in writing. As a consequence, researchers cannot detect the relationship between words following traditional grammar rules. Furthermore, some texts on the Internet do not use traditional punctuations and stop words as separators for meaning groups. Thus, punctuations like a comma, period and exclamation point and stop words may not mean what they mean in traditional writing. The author considers them as one kind of noise and removes them from data. Then, online texts have many special symbols such as '@' and '#'. They are useful in these texts and have their functions. However, they hardly affect the sentiment embedded in texts. Therefore, the author considers these symbols as noises (O'Hare et al., 2009). Finally, in some special texts, there are some frequently used words but do not affect the sentiment. For example, in tweets about airlines in the US airline sentiment dataset, there are many nouns related to airline names. The author considers these nouns as noises as well because they would be mistaken by the classifier as very important features.

Third, tokenization should be applied to the data and labels. This step plays a significant role because classifiers only accept numerical data stored in vectors, matrixes or tensors as inputs. The author firstly sorts words in texts according to the frequency of them in datasets. Then, the author would only keep some of the most frequently occurring words to reduce the negative impact of sparsity. Next, words are converted into numerical data, and these numbers represent the frequency of their occurrence. Words with higher frequency get smaller numbers as their representatives. Finally, each unit is expressed as a series of numbers, stored in a computer as a list.

Fourth, the one-hot is applied to data. It is a basic and simple technique to capture the presence of information of words in data. Its principle can be explained as follows:

each word has a fixed position in the entire unit represented by a vector. If a word is represented by '1' in the tokenization process, and the word appears in a unit. Then, in the first position in this unit (the vector), there would be a '1'. However, if the word does not exist in this unit, the first position would have a value '0'. Labels are also transferred as numbers. 'Positive' is represented by '1' and 'negative' is represented by '0'. If there are three kinds of labels in the dataset, labels would be represented just like one-hot. 'Positive' is represented by '1,0,0', 'neutral' is represented by '0,1,0' and 'negative' is represented by '0,0,1'.

The literature review highlighted that there are many feature selection approaches before classification tasks. Most of them are conducted based on statistical principles. These feature selection approaches belong to three types -- filter, wrapper, and embedded techniques. The filter is about ranking before classification, the wrapper is about evaluating a subset of features, and embedded techniques are considered as a part of the training. Although feature selection can reduce the size of input vectors and improve the accuracy of classification by choosing the most important features before classifying, it is not the main point in this research. Moreover, applying feature selection would take a lot of time, especially on large datasets. Consequently, this research only takes the presence and frequency of words as features but no more complicated feature selection.

3.6 DEEP NEURAL NETWORK CLASSIFICATION

Classification using deep neural networks is the core phase of this research. Three deep neural networks – LSTM, DBN, and DELM would be used as classifiers in this research for sentiment analysis. The preparation has been introduced in section 3.4 Datasets Selection and section 3.5 Text-based Data Pre-processing. The principles of the three deep neural networks are introduced in detail in section 2.4 Deep Neural Network Approach in the literature review chapter. Before training, each dataset should be split into a training set, validation set, and test set. When training, the training set would be used first to build the deep learning model. Then the validation set would be sent to the model for prediction, and this would help calculate the accuracy of the model. Next, the author would adjust all the hyper parameters and train a new model on the training set iteratively to make sure the deep neural network has the highest accuracy on the validation set. At last, the author would train the model on the training set using the best hyper parameters. After training, the test set would be sent to the best

model for prediction. Training time and accuracy would be recorded as results. Regularly, the training set accounts for 70%, the validation set accounts for 20%, and the test set accounts for 10%. This research would use this ratio on all six datasets.

3.6.1 Word Embedding

Three deep neural networks have the same inputs but the training process would be slightly different. First, as mentioned in section 2.4.1, LSTM is designed for sequential data. Thus, the inputs should be arranged in sequences. However, one-hot makes a big vector for each unit which is not sequential, so that there should be a data conversion layer between the LSTM layer and inputs. Word embedding is a suitable technique that can solve the problem. It would create an independent vector for each word in the unit so that the original input vectors would change into input matrixes after applying word embedding (Mikolov et al., 2013). Word embedding can be considered as a hidden layer or a lookup table as shown in Figure 3-3.

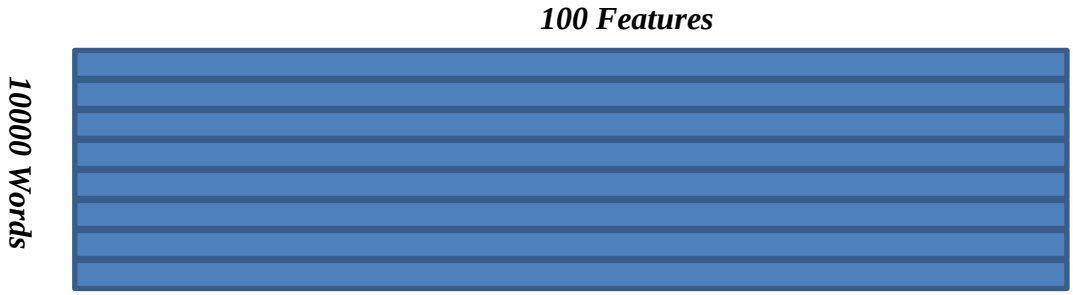

Figure 3-3 Word Embedding (Mikolov et al., 2013)

When training LSTM, it would perform as a hidden layer above LSTM layers. However, this technique cannot be applied with DBN and DELM currently because these two deep neural networks do not accept a matrix as an input unit. They are not designed for sequential data. If the word embedding layer is forcibly placed into models of these two algorithms, the models would lose the ability to predict correctly because the formats of features are wrong.

3.6.2 LSTM Training

The training process of LSTM can be easily conducted by a deep learning library -- Keras (Chollet, 2015). The training model includes three layers – embedding layer, LSTM layer and dense layer (Tang et al., 2015a).

In the embedding layer, word embedding is conducted to relieve the problem of sparsity. There are three hyper parameters in this layer. The first one is ‘max words’ which represents how many words with the highest frequency would be selected for training as mentioned in the tokenization step in section 3.5. The second one is ‘embedding size’ which determines the size of the vector for each word after word embedding. The third one is ‘max length’ which means the size of each unit (how many words a unit has).

In the LSTM layer, there are two hyper parameters – ‘LSTM parameter’ and ‘dropout rates’. ‘LSTM parameter’ decides what shape the LSTM layer would be like for each time-step as mentioned in section 2.4.1. ‘Dropout rates’ indicates how many ratios of weights would be randomly removed, which would be explained in section 3.6.5.

In the dense layer, there are also two hyper parameters – ‘dense nodes’ and ‘activation function’. The dense layer is the abbreviation of the densely-connected neural network layer. This layer is the carrier of activation functions which would be explained in section 3.6.6. ‘Dense nodes’ should match the types of labels, and ‘activation function’ can be chosen from softmax, elu, selu, softplus, softsign, relu, tanh, sigmoid, hard_sigmoid, exponential and linear. This layer is designed as the final classifier of the model.

When compiling, some other hyper parameters can control how the program would run. First, ‘optimizer’ is an important hyper parameter that can determine which trick would be applied in back-propagation. This would be described in detail in section 3.6.7. Second, ‘loss’ indicates which loss function would be used in the model. This would be explained in section 3.6.7. Third, ‘epochs’ describes how many loops would the training requirements for training the model through the entire training set. Fourth, ‘batch_size’ gives the size of the training subset in one back-propagation process.

3.6.3 DBN Training

The training process of DBN can be implemented by using open source codes developed by ‘albertbup’ on GitHub (albertbup, 2017). The training model also includes many hyper parameters (Ruangkanokmas et al., 2016). First, ‘hidden_layers_structure’ is a significant hyper parameter that can determine the

structure applied in model training. Commonly, there are two layers in the model, but the nodes in each layer can be adjusted. Second, 'learning_rate_rbm' indicates the learning rate of the unsupervised learning process in DBN. Third, 'learning_rate' describes the learning rate of the supervised learning process in DBN. This is related to back-propagation which would be explained in section 3.6.7. Fourth, 'n_epochs_rbm' gives how many loops needed for training the model through the entire training set in unsupervised learning part. Fifth, 'n_iter_backprop' indicates how many loops needed for training the model through the entire training set in the supervised learning part. Sixth, 'batch_size' gives the size of the training subset in one back-propagation process. Seventh, 'activation_function' can be chosen from softmax, relu, tanh and sigmoid. Eighth, 'dropout_p' indicates how many ratios of weights would be randomly removed, which would be explained in section 3.6.5.

3.6.4 DELM Training

The training process of DELM can be implemented by using open source codes developed by Tang et al. (2015b). Before training the DELM model, data including training set, validation set, and test set should be transferred from NumPy array to lists in MATLAB. The training model also includes many hyper parameters (Roul et al., 2017). First, 'TotalLayers' plays an important role as a hyper parameter that can determine numbers of hidden layers in model training. Regularly, there are two layers in the model. Second, 'HiddenNeurons' indicates the number of nodes in each layer. The remaining four hyper parameters are all designed for adjusting the weights in the model.

3.6.5 Dropout

The deep neural network has a strong fitting ability. However, overfitting always exists when training deep learning models. Overfitting means weights in the model fit data and labels in training data at a high level, so that the model may have good accuracy on the training set, but it cannot perform well on the validation set and test set. Researchers found that randomly removing nodes in the model can lead the model to learn more meaningful features. This improves the generalization ability of the model to new data and makes the model robust (Hinton et al., 2012).

3.6.6 Activation Function

In deep learning, the activation function is the last step in node internal calculation. Before this step, there is a linear calculation to multiple weights and inputs. This kind of function is designed to mimic biological neurons. When a neuron is stimulated, it would choose to pass or not pass the stimulus, and the degree of stimulation will vary when passing. Thus, the activation function in one node determines how to pass the value calculated by linear calculation to the next node (Goodfellow et al., 2016).

There are a lot of activation functions as listed in section 3.6.2. However, only some of these nonlinear functions are commonly used. First, the sigmoid function is often used for classification tasks, and the author uses it in the dense layer in the LSTM model. It can be explained as the following equation.

$$f(x) = \sigma(x) = \frac{1}{1 + e^{-x}}$$

When the inputs are vectors, the sigmoid function would evolve into softmax function which can be explained as the following equation. The author uses it when dealing with the US airline sentiment dataset because labels in it are vectors but not numbers.

$$f_i(\boldsymbol{x}) = \frac{e^{\boldsymbol{x}_i}}{\sum_{j=1}^{J} e^{\boldsymbol{x}_j}} \; for \; i = 1, \dots, J$$

For some densely connected deep neural networks, the sigmoid function may impact back-propagation and cause a problem of gradient vanishing or exploding. Thus, the rectified linear unit is introduced to solve the problem. It is described as the following equation. This activation function is used in DBN.

$$f(x) = \begin{cases} 0 \;\; for \; x < 0 \\ x \;\; for \; x \geq 0 \end{cases}$$

3.6.7 Back-propagation

Back-propagation is the essence of deep neural network training. It is used for fine-tuning weights in nodes based on the error rate in loss function (section 3.6.8) got in the previous epoch. It makes the model robust and has better generalization ability (Goodfellow et al., 2016).

When the training starts, the model would let data in training set go through every node in it and get the value of the loss at the end. This process is called feed-forward. Back-propagation is the opposite process of feed-forward since it feeds the loss backward and adjusts weights.

The function to conduct back-propagation is an optimization function. Commonly, the optimization function is gradient descent. However, traditional gradient descent has a weakness when dealing with a large amount of data. Thus, researchers change it to other variants sometimes. For example, in this research, the author uses 'rmsprop' in the LSTM model for optimizer in back-propagation.

Deep neural networks have multiple layers. There are many functions in each layer. Therefore, feeding backward equals to doing a series of partial derivatives of those functions. Then, deltas are calculated and weights are updated.

3.6.8 Loss Function

The loss function is significant in deep learning because it makes theoretical algorithms more practical and easier to optimize (Goodfellow et al., 2016). The value of it can be considered as the gap between expected prediction and actual prediction. Back-propagation cannot be done if not establishing effective loss function.

There are many types of loss functions. Mean squared error (MSE) is usually used as a basic loss function because it is easy to understand and conduct. Furthermore, it works well in most cases. It is described as the following equation. y' is the prediction, y is the ground truth and Y is the size of the dataset.

$$MSE = \frac{(y' - y)^2}{Y}$$

Cross-entropy loss is also frequently used. Being very wrong and being very confident would be punished by it. The following equation can describe it in detail.

$$Cross\ Entropy\ Loss = -(ylog(p) + (1 - y)\log(1 - p))$$

When training the LSTM model, the author uses categorical cross entropy for US airline sentiment dataset and binary cross entropy for other datasets.

3.6.9 Summary of Deep Neural Network Classification

In section 3.6 Deep Neural Network Classification, the author introduces how the sentiment classification would be implemented. Details about training are

demonstrated and related techniques are indicated such as word embedding, dropout, activation function, back-propagation, and loss function.

3.7 PERFORMANCE EVALUATION

There are two core results should be collected and analysed in this research – accuracy and training time. The author would conduct experiments using three deep neural networks upon six datasets so that there are 18 different situations. The author does 10 the same experiments for each situation to make the results more convincing.

Also, statistical approaches should be applied to these results to obtain effective conclusions. First, confidence intervals are used to quantify the performance of deep learning models. It can provide bounds on the mean of a population. When applying on classification models, it would indicate a 95% likelihood that the given range covers the error of the model (95% is frequently used). This approach can be described as the following equation (Ghasemi and Zahediasl, 2012). When choosing a 95% confidence level, z is 1.96.

$$Interval = z * sqrt(\frac{(error * (1 - error))}{examples})$$

Second, statistical tests such as the Shapiro-Wilk test, t-test, and Mann–Whitney U test should be applied to compare the difference in mean between results of three deep neural networks on different datasets. Shapiro-Wilk test can judge whether the data in the dataset follows the normal distribution. This would determine which test should be used for comparison. If the data follows the normal distribution, a t-test would be used. Otherwise, the Mann-Whitney U test would be used in this research (Ghasemi and Zahediasl, 2012).

3.8 SUMMARY

In conclusion, after designing methodology on sentiment analysis, the author finds that this research aims to achieve one main objective: comparison of training time and accuracy between LSTM, DBN, and DELM approaches on sentiment analysis.

This chapter selects appropriate datasets, chooses suitable pre-processing steps for text-based data and makes classification and evaluation clear. In the next chapter,

there would be some analysis of datasets, pre-processing results, classification results, and an explanation for them.

Chapter 4: Experimental Results and Discussion

Chapter 4 describes the implementation of experiments designed in the methodology chapter. In this chapter, the author details all the results of this research and provides some analysis of the results with interpretation, inference, and evaluation. The results are linked inextricably to the design – describe what happened factually. The analysis focuses on discussing the 'meaning' of the results and the author's 'finding'.

Section 4.1 is related to datasets selection and some basic statistical descriptions of datasets. Section 4.2 indicates the results of text-based data pre-processing. Section 4.3 details the implementation of deep neural network classification. Section 4.4 gives statistical tests on experimental results and offers some explanation.

4.1 DATASETS SELECTION

As described in the methodology chapter, the datasets selection phase discovers six datasets which are distinctive and easy to process. The reasons for choosing them and the sample data in them are shown in section 3.4 in the methodology chapter. In this chapter, detailed statistics about the six datasets would be displayed.

Table 4-1 Statistics of Datasets Used in This Research

Dataset name	Total number of polarities	Total number of units	Total size of the dataset
IMDB review	2	25000	33.1MB
Amazon review	2	3600000	1.6GB
Hotel review	2	515738	238.2MB
US airline sentiment	3	14640	3.4MB
Twitter	2	1600000	238.8MB
Reddit	2	1010826	255.3MB

As shown in Table 4-1, the Amazon review dataset holds the largest number of units and the largest storage space. The Twitter dataset and the Reddit dataset have millions of units, too. The Hotel review dataset has half a million units, but the storage space assigned for it is similar to the Twitter dataset and the Reddit dataset. This indicates that each unit in the Hotel review dataset contains more data. The IMDB review dataset and the US airline sentiment dataset contain only tens of thousands of units. However, the US airline sentiment dataset is special because texts in it can be divided into three polarities.

The IMDB review dataset has long sentences about people’s sentiment after watching movies. Texts in this dataset contain a lot of subjective sentiment because they are the reviewer's personal suggestion for the movie. Therefore, this dataset is valuable in this research. The Amazon review dataset has a large collection of online shoppers' messages for the items they have purchased. These messages also contain a lot of subjective sentiment, so it is suitable for this study. The Hotel review dataset

gets travellers' evaluation of multiple hotels. These reviews are based on their true feelings, so they contain a lot of subjective sentiment. For the US airline sentiment dataset, passengers poured their complaints or satisfaction with the airlines into texts. In the Twitter dataset, many kinds of sentiment occur in daily life. In the Reddit dataset, participants in the online forum mixed irony in their comments.

For each dataset, there are enough texts for further processing and classification. Moreover, each dataset has its characteristics, and the language style of texts in them is different. In deep learning, data is important because it affects which features would the model learn so that it determines the prediction capability of the model. In this research, the author finds six appropriate datasets for training deep learning algorithms on sentiment analysis showed above. This is essential for the experiments and helpful for getting convincing results.

4.2 TEXT-BASED DATA PRE-PROCESSING

As mentioned in the methodology chapter, the text-based data pre-processing phase pre-processes data in six selected datasets before classification. The steps for this phase and the reasons for implementing these steps are shown in section 3.5 in the methodology chapter. In this chapter, detailed statistics and the results after these steps would be displayed.

Table 4-2 Selection of Raw Data

Dataset name	Total number of selected units
IMDB review	25000
Amazon review	100000
Hotel review	110990
US airline sentiment	14640
Twitter	100000
Reddit	100000

As shown in Table 4-2, experiments in this research do not use all the data in the six datasets. Pre-processing and classification cost a lot of computing power and need time. Since the experimental computer used in this research has limited computing power, if using a dataset containing millions of sentences, the experiments would take a lot of time. Thus, the author only intercepts nearly 100,000 units of all large datasets. The Amazon dataset, the Hotel review dataset, the Twitter dataset, the Reddit dataset are affected by this. The IMDB review dataset and the US airline sentiment dataset contain only tens of thousands of units. Therefore, they keep all the content in themselves for pre-processing.

There is another problem with the raw data – imbalance. Most of the data from the Internet is unbalanced, which is unavoidable. Unbalanced data would make the feature extraction of the model biased in classification tasks, resulting in a bias in the classification results. Since the author manually selected the raw data in this research, the data can be generally balanced as displayed in the following figure.

Figure 4-1 Polarity Ratio in Six Datasets

Except that the US airline sentiment dataset cannot be balanced while taking into account not removing data, the number of positive labels is equivalent to the number of negative labels in the other five datasets. The IMDB review dataset has 12500 positive labels and 12500 negative labels. The Amazon review dataset contains 51267 units labelled as positive and 48733 units labelled as negative. There are 61149 positive labels and 49841 negative labels in the Hotel review dataset. The US airline sentiment dataset owns 3099 positive labels, 9178 negative labels, and 2363 neutral labels. The Twitter dataset holds 100000 units, half labelled as positive and half labelled as negative. The Reddit dataset retains 42310 units that contain irony and 57690 units that get common texts. After selecting particular data in the six datasets, the sample raw data can be seen in the following table.

Table 4-3 Sample of Raw Data

Dataset name	Sample data
IMDB review	"Carriers" follows the exploits of two guys and two gals in a … you really don't care what happens to anybody.
Amazon review	Remember, Pull Your Jaw Off The Floor After Hearing it: … just poured his heart on and wrote it down on paper.
Hotel review	You When I booked with your company on line you showed … I wasn t Not happy and won t be using you again
US airline sentiment	@VirginAmerica and it's a really big bad thing about it
Twitter	@LOLTrish hey long time no see! Yes.. Rains a bit ,only a bit LOL , I'm fine thanks , how's you ?
Reddit	I could use one of those tools.

Raw data has multiple kinds of noises such as meaningless punctuations, symbols, stop words, subjects, and capitalization. These noises can affect feature extraction, thereby weakening the ability of the model to make correct classification predictions. Thus, noise removal should be implemented, and the results are demonstrated in the following table.

Table 4-4 Sample Data after Noise Removal

Dataset name	Sample data
IMDB review	carriers follows the exploits of two guys and two gals in a … really don't care what happens to anybody
Amazon review	remember pull your jaw off the floor after hearing it … just poured his heart on and wrote it down on paper
Hotel review	booked company line showed … happy using
US airline sentiment	really big bad thing
Twitter	hey long time see yes rains bit bit lol fine thanks
Reddit	could use one tools

After removing noises, data becomes clean and ready for tokenization. Although each unit does not look like a complete sentence after this step, its meaning, especially the sentiment information contained in it, is still preserved. To make the computer recognize the data, the text data must be converted to numbers. This step is tokenization that is introduced in section 3.5 in the methodology chapter. The sample results are displayed in the following table.

Table 4-5 Sample Data after Tokenization

Dataset name	Sample data
IMDB review	[553 12 5844 23 81 30 3 1547 2 33 416 5 … 63 89 456 48 568 5 1810]
Amazon review	[0 0 0 0 0 0 0 0 0 0 0 0 0 0 … 35 8109 55 548 19 2 768 7 171 19 706]
Hotel review	[2477 197 1425 741 … 257 533]
US airline sentiment	[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2546 174 137 8]
Twitter	[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 98 69 10 27 64 2691 161 161 14 382 31]
Reddit	[0 0 0 0 0 0 0 0 0 0 0 20 48 6 3212]

A large number of ‘0’ in tokens represent placeholders. In the IMDB review dataset, the Amazon review dataset, the Hotel review dataset, each unit has many words. The sizes of some units are too big for classification. As we knew, in a single paragraph, the sentiment tendency is generally uniform, so the author intercepts a fixed-length token for each unit. As long as this length is appropriate, there would be no loss of sentiment information. For the IMDB review dataset, the Amazon review dataset, and the Hotel review dataset, this length is 500, 200, and 20. For the US airline sentiment dataset, the Twitter dataset, and the Reddit dataset, this length is consistent with the longest unit length in each dataset, which is 26, 35, and 15. In brief, the units in the same dataset are processed as tokens of the same size.

However, this series of processing has some side effects, and some units become 0 after the content has been deleted. Thus, the author deletes these units and their labels because keeping them would lead to errors in feature extraction. The numbers of deleted units are shown in the following table.

Table 4-6 Deletion of Data

Dataset name	Total number of deleted units
IMDB review	0
Amazon review	0
Hotel review	467
US airline sentiment	0
Twitter	868
Reddit	3622

The final step is converting tokens to one-hot form. For the Hotel review dataset, the author creates a 5000-dimensional vector for each unit. For the other five datasets, the author creates a 10000-dimensional vector for each unit because the level of the linguistic diversity of texts in the Hotel review dataset is lower than in the other datasets. Although these vectors are very sparse, this problem is caused by the sparseness of natural language and is one of the problems that researchers in the NLP field are currently trying to solve. Such a feature selection approach is not the

best one, but it is suitable and generally fair for the three deep neural networks in this research. Sample one-hot data is displayed in the following table.

Table 4-7 Sample Data after One-hot

Dataset name	Sample data
IMDB review	[1. 1. 1. ... 0. 0. 0.]
Amazon review	[1. 1. 0. ... 0. 0. 0.]
Hotel review	[0. 0. 0. ... 0. 0. 0.]
US airline sentiment	[0. 0. 0. ... 0. 0. 0.]
Twitter	[0. 0. 0. ... 0. 0. 0.]
Reddit	[0. 0. 0. ... 0. 0. 0.]

4.3 DEEP NEURAL NETWORK CLASSIFICATION

This research is looking for the strengths and weakness of DELM by comparing its classification capability in sentiment analysis to LSTM and DBN based on the same datasets, similar data pre-processing techniques, and identical operating environment. After processing in the previous sections, data can be directly used to train deep learning models. Data in six datasets should be divided into the training set, the validation set, and the test set. Word embedding without pre-training parameters would be used as a part of the LSTM model as mentioned in section 3.6 in the methodology chapter. The DELM model would be trained in MATLAB, but data would be the same when training, validating and testing three models.

The training, validating, and testing hardware and software conditions for LSTM, DBN, and DELM are listed as follows: Laptop, Intel-i5 2GHz CPU, 16G DDR3 RAM, macOS Mojave, Python 3.6, Numpy 1.14.3, Pandas 0.23.0, Python Regular Expression Operations 2.2.1, Natural Language Toolkit 3.3, Sklearn 0.19.1, Keras 2.2.4, MATLAB R2019a.

After training LSTM models, DBN models, and DELM models several times with different hyper parameters on different datasets, the author gets the best sets of hyper parameters for 18 different situations (3 deep neural networks and 6 datasets). Each optimal set of hyper parameters is concluded based on the prediction accuracy of

the model on the validation set. These sets of hyper parameters are demonstrated in the following three tables.

Table 4-8 The Best Sets of Hyper Parameters for LSTM

Hyper parameter	Dataset					
	IMDB review	Amazon review	Hotel review	US airline	Twitter	Reddit
Max words	10000	10000	5000	10000	10000	10000
Embedding size	32	64	64	32	128	8
Max length	500	200	20	26	35	15
LSTM parameter	32	64	64	32	128	8
Dropout rates	0.05	0.05	0.3	0.2	0.5	0.05
Dense nodes	0.05	0.05	0.3	0.2	0.5	0.05
Epochs	3	4	3	6	3	4
Batch_size	128	128	128	64	128	256

Table 4-9 The Best Sets of Hyper Parameters for DBN

Hyper parameter	Dataset					
	IMDB review	Amazon review	Hotel review	US airline	Twitter	Reddit
Hidden_layers_structure	16, 16	16, 16	16, 16	256, 256	16, 16	16, 16
Learning_rate_rbm	0.01	0.05	0.05	0.05	0.05	0.05
Learning_rate	0.1	0.1	0.1	0.1	0.1	0.1
N_epochs_rbm	10	10	10	10	10	10
N_iter_backprop	20	10	10	10	10	10

Batch_size	256	128	64	128	128	16
Dropout_p	0.1	0.05	0.05	0.05	0.1	0.1

Table 4-10 The Best Sets of Hyper Parameters for DELM

Hyper parameter	**Dataset**					
	IMDB review	**Amazon review**	**Hotel review**	**US airline**	**Twitter**	**Reddit**
TotalLayers	2	2	2	2	2	2
HiddenNeurons	9000, 10000	9000, 10000	9000, 10000	5000, 10000	9000, 10000	9000, 10000
C1	1, 1, 1e1	1e6, 1e-2, 1e6	1e6, 1e-2, 1e6	1e2, 1e-2, 1e4	1e2, 1e-2, 1e4	1e2, 1e-2, 1e4
RhoValue	0.05	0.05	0.1	0.4	0.5	0.5
Sigpara	1,1	1,1	1,1	1,1	1,1	1,1
Sigpara1	5,5	5,5	5,5	5,5	3,3	3,3

Since the limitation in the experimental computer's power, in DBN and DELM training, the author focuses on DBN and DELM with two hidden layers. A small number of experiments have shown that DBN and DELM with three hidden layers do not increase the prediction accuracy in addition to more time. So all the best sets of hyper parameters fall on DBN and DELM with two hidden layers.

In the next section, the author would use 18 sets of the best hyper parameters to retrain deep learning models, and the training time would be collected as a part of the experimental results. At the end of the training, the model would be used to make predictions on the test sets, and the accuracy would be collected as another part of the experimental results.

4.4 PERFORMANCE EVALUATION AND DISCUSSION

In this section, the most important results would be demonstrated. The training time and accuracy would be displayed, and multiple comparisons would be implemented. The author does 10 the same experiments for each situation to make the results more convincing. Thus, in the following tables and figures, the training time and accuracy are represented by the mean of the ten results in each situation.

4.4.1 Results of LSTM

The mean of the training time of the LSTM models is presented in Figure 4-2. The x-axis represents different datasets. The y-axis represents the time in seconds. The blue areas represent the training time. More details are shown in Table 4-11.

The mean and 95% confidence interval of the accuracy of testing on six datasets using LSTM as the classifier is displayed in Figure 4-3. The x-axis represents different datasets. The y-axis represents the accuracy. The grey line represents the mean accuracy. The light blue areas mean the 95% confidence interval of the mean of the ten accuracy values. More details are shown in Table 4-11.

Table 4-11 Results of LSTM

Data type	**Dataset**					
	IMDB review	**Amazon review**	**Hotel review**	**US airline**	**Twitter**	**Reddit**
Training time (s)	228.3	619.4	57.79	25.25	198.2	20.23
Accuracy	0.8591	0.9041	0.9313	0.7918	0.7505	0.6687
95% CI lower limit for accuracy	0.846	0.898	0.926	0.771	0.742	0.659
95% CI higher limit for accuracy	0.873	0.91	0.936	0.812	0.759	0.678

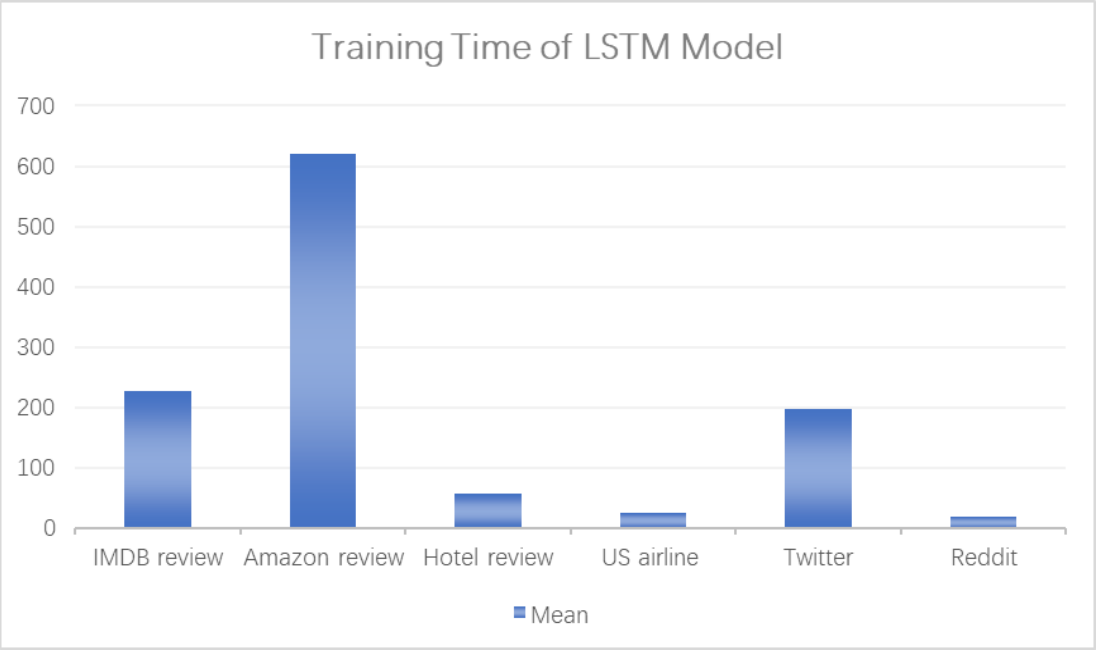

Figure 4-2 Training Time of LSTM Model

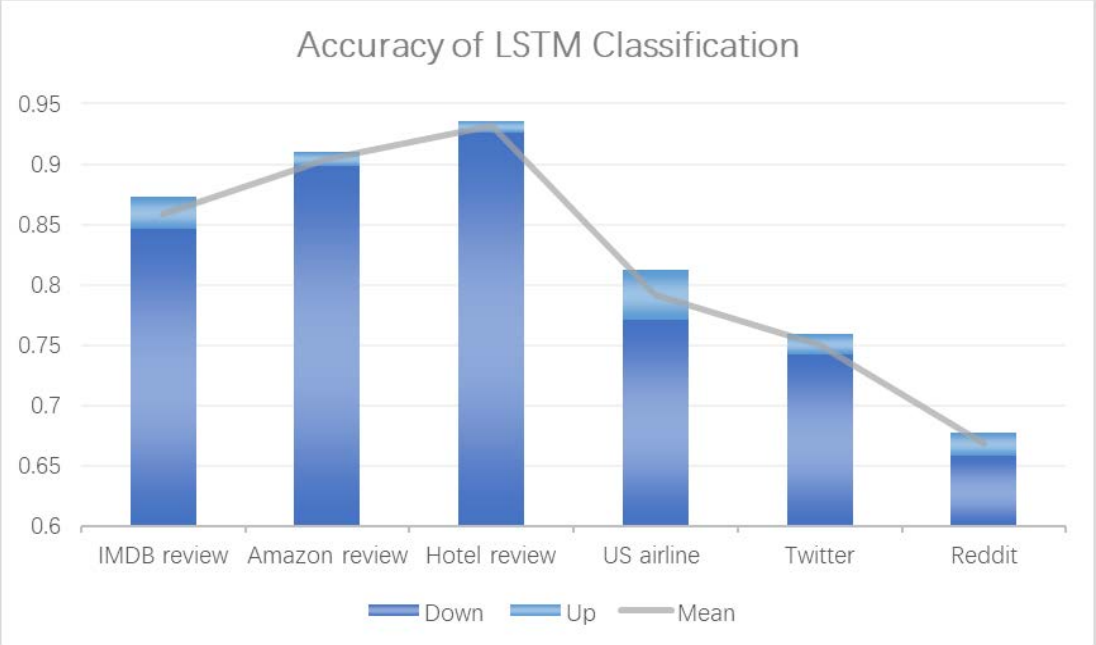

Figure 4-3 Accuracy of LSTM Classification

As presented in the above table and figures, the LSTM algorithm performs differently on different datasets. According to Figure 4-2, the LSTM model takes the longest training time on the Amazon review dataset. Moreover, the training on the Reddit dataset takes the shortest time. In the author's opinion, the dimensions of the word embedding and the number of layers inside the LSTM are the main factors determining the training time. When conducting statistical tests on every pair of data

about the training time of LSTM models, the significant difference in mean between each pair of data never gets the 1% level (refer to tables in Appendix A). Thus, the above conclusions can be confirmed.

Due to Figure 4-3, the LSTM model achieves the best prediction accuracy on the Hotel review dataset. Apart from this, the model grants the worst prediction accuracy on the Reddit dataset. In the author's opinion, the quality of the data and labels in the datasets are the main factors that influence the prediction accuracy. When conducting statistical tests on every pair of data about the prediction accuracy of LSTM models, the significant difference in mean between each pair of data never achieves the 1% level (refer to tables in Appendix A). Thus, the above conclusions can be approved.

4.4.2 Results of DBN

The mean of the training time of the DBN models is presented in Figure 4-4. The x-axis indicates different datasets. The y-axis indicates the time in seconds. The blue areas indicate the training time. The mean and 95% confidence interval of the accuracy of testing on six datasets using DBN as the classifier is displayed in Figure 4-5. The x-axis indicates different datasets. The y-axis indicates the accuracy. The grey line indicates the mean accuracy. The light blue areas mean the 95% confidence interval of the mean of the ten accuracy values. More details are shown in Table 4-12.

Table 4-12 Results of DBN

Data type	**Dataset**					
	IMDB review	**Amazon review**	**Hotel review**	**US airline**	**Twitter**	**Reddit**
Training time (s)	452.5	1608.9	896.3	2376	1563.8	1525
Accuracy	0.8922	0.8783	0.9249	0.7717	0.7437	0.6568
95% CI lower limit for accuracy	0.88	0.872	0.92	0.75	0.735	0.647
95% CI higher limit for accuracy	0.905	0.885	0.93	0.793	0.752	0.666

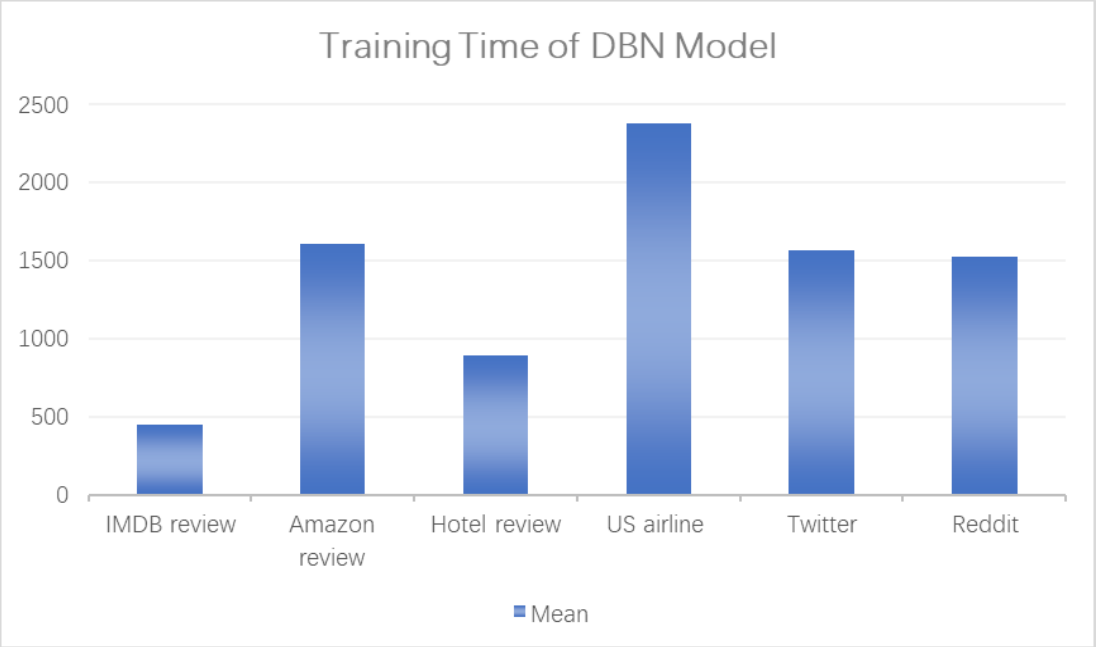

Figure 4-4 Training Time of DBN Model

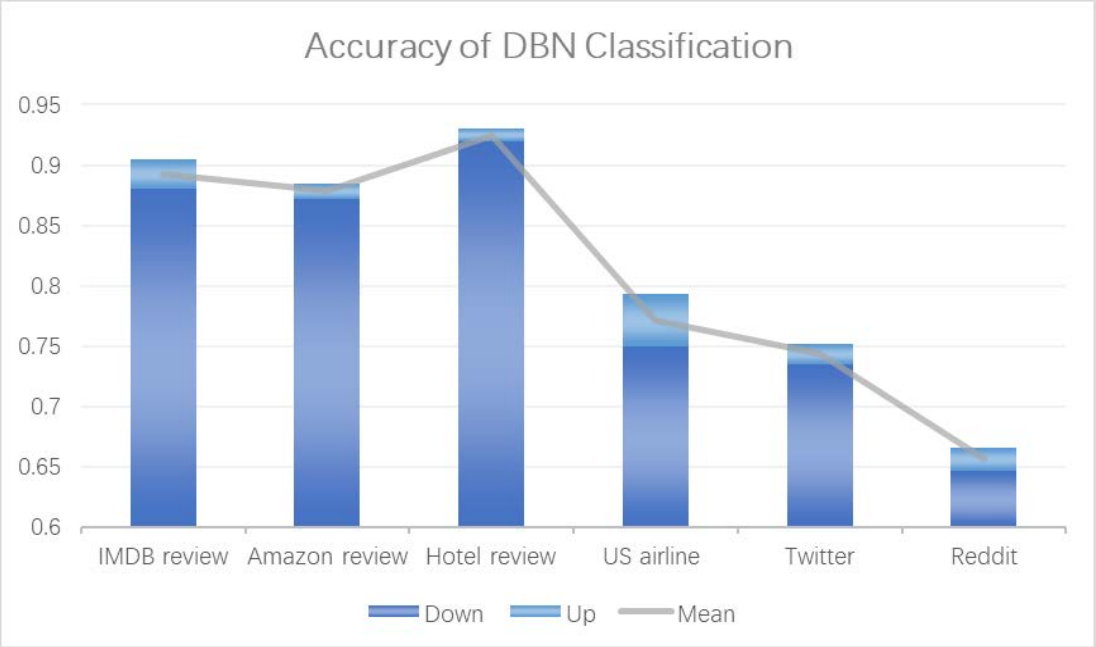

Figure 4-5 Accuracy of DBN Classification

As presented in the above table and figures, the DBN algorithm gets different experimental results on different datasets. According to Figure 4-4, the DBN model takes the longest training time on the US airline sentiment dataset. Additionally, the training on the IMDB review dataset takes the shortest time. In the author's opinion, the structure of hidden layers in the DBN model and the size of batch in back-propagation are the main factors affecting the training time. When conducting

statistical tests on every pair of data about the training time of DBN models, the significant difference in mean between each pair of data never gets the 1% level (refer to tables in Appendix A). Thus, the above conclusions can be robust.

Due to Figure 4-5, the DBN model achieves the best prediction accuracy on the Hotel review dataset. Furthermore, the model grants the worst prediction accuracy on the Reddit dataset just like that the results of implementing LSTM. Again, the quality of the data and labels in the datasets should be the main factors that affect the prediction accuracy. In statistical tests, the significant difference in mean between each pair of data about DBN prediction accuracy also never achieves the 1% level (refer to tables in Appendix A). Thus, the above conclusions can be robust.

4.4.3 Results of DELM

The mean of the training time of the DELM models is displayed in Figure 4-6. The x-axis indicates different datasets. The y-axis indicates the time in seconds. The blue areas indicate the training time. The mean and 95% confidence interval of the accuracy of testing on six datasets using DELM as the classifier is displayed in Figure 4-7. The x-axis indicates different datasets. The y-axis indicates the accuracy. The grey line indicates the mean accuracy. The light blue areas mean the 95% confidence interval of the mean of the ten accuracy values. More details are shown in Table 4-13.

Table 4-13 Results of DELM

Data type	**Dataset**					
	IMDB review	**Amazon review**	**Hotel review**	**US airline**	**Twitter**	**Reddit**
Training time (s)	1544.3	3111.1	2563	436.3	3088.2	3052.4
Accuracy	0.8813	0.8692	0.9270	0.7676	0.7384	0.6707
95% CI lower limit for accuracy	0.869	0.863	0.922	0.746	0.73	0.661
95% CI higher limit for accuracy	0.894	0.876	0.932	0.789	0.747	0.68

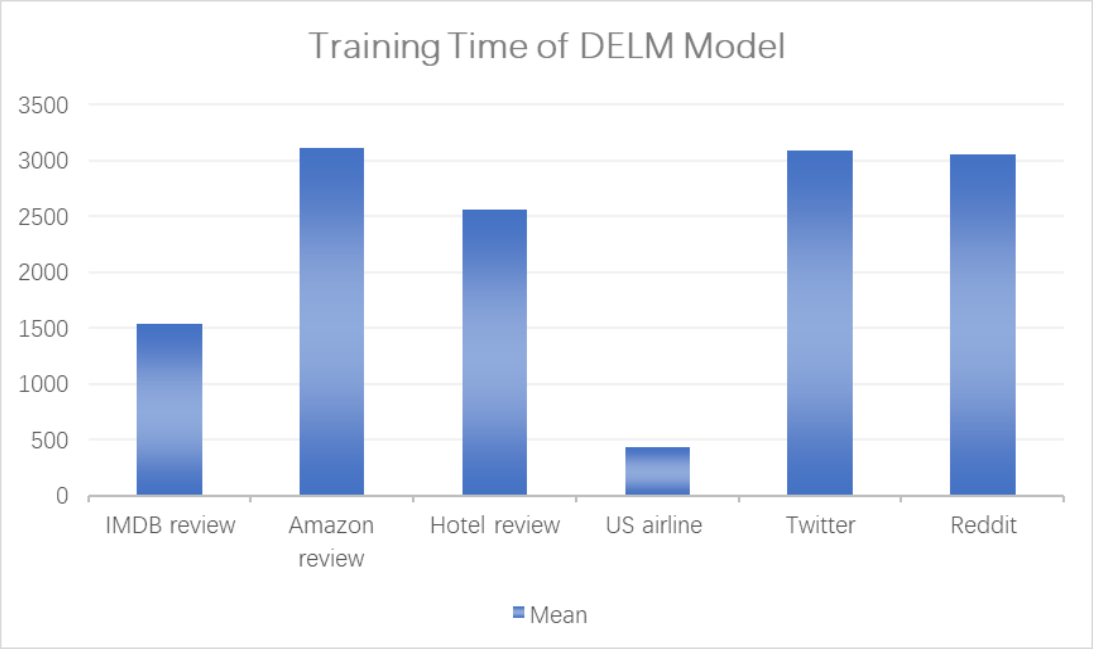

Figure 4-6 Training Time of DELM Model

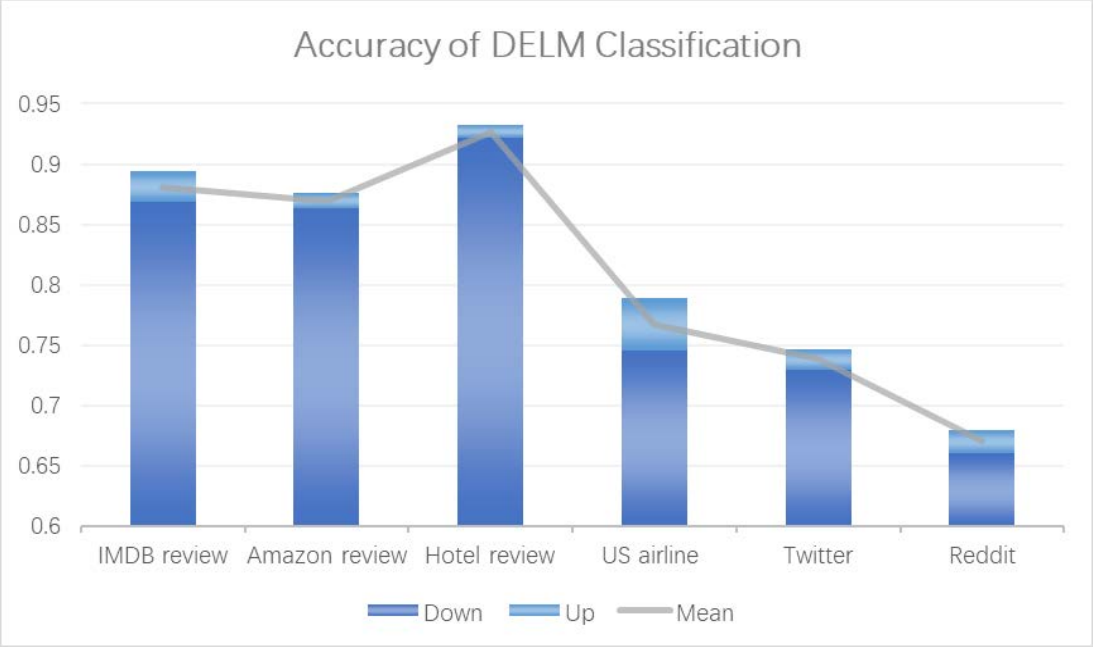

Figure 4-7 Accuracy of DELM Classification

As presented in the above table and figures, the DELM algorithm generates different results on different datasets. According to Figure 4-6, the DELM model takes the longest training time on the Amazon review dataset. Moreover, the training on the US airline sentiment dataset takes the shortest time. In the author's opinion, the structure of hidden layers in the DELM model and the size of the dataset are the main factors determining the training time. When conducting statistical tests on every pair

of data about the training time of DELM models, the significant difference in mean between training time on the Amazon review dataset and the Twitter dataset achieves the 5% level. Moreover, the significant difference in mean between training time on the Amazon review dataset and the Reddit dataset achieves the 5% level, and the significant difference in mean between training time on the Twitter dataset and the Reddit dataset also achieves the 5% level. Thus, the training time of DELM on these three datasets can be considered as the same. However, the significant difference in mean between the other pairs of data never gets the 1% level (refer to tables in Appendix A). Thus, this evidence supports a part of the above conclusions.

Due to Figure 4-7, the DELM model achieves the best prediction accuracy on the Hotel review dataset. Apart from this, the model grants the worst prediction accuracy on the Reddit dataset just like that the results of implementing LSTM and DBN. Therefore, the quality of the data and labels in the datasets are the main factors that influence the prediction accuracy. When conducting statistical tests on every pair of data about the prediction accuracy of LSTM models, the significant difference in mean between each pair of data never achieves the 1% level (refer to tables in Appendix A). Thus, the above conclusions can be supported.

4.4.4 Comparisons

Since this research has the aim of detecting the strengths and weakness of DELM, comparisons on the results of LSTM, DBN, and DELM should be essential. In Figure 4-8. The x-axis indicates different datasets. The y-axis indicates the time in seconds. The training time for LSTM, DBN and DELM are represented as blue, orange and grey, respectively. In Figure 4-9. The x-axis indicates different datasets. The y-axis indicates the accuracy. The prediction accuracy for LSTM, DBN, and DELM are represented as dark blue, grey and light blue, respectively.

It is obvious that LSTM takes the shortest training time on every dataset. On most datasets, DELM takes the longest training time. In the author’s view, the reason for less training time when using LSTM is that there is a word embedding layer in the LSTM model, and it can significantly reduce the dimensions of the model's input vectors. The reason why DELM spends a lot of training time can be explained in this way. DELM runs in MATLAB, which takes up a lot of memory and slows down the training speed. Furthermore, DELM uses matrix operations to update internal weights. If the training set is large, the training speed would be slower than back-propagation.

When conducting statistical tests on every pair of data about the training time of LSTM, DBN, and DELM models on the same dataset, The significant difference in mean between each pair of data never gets the 1% level (refer to tables in Appendix A). Thus, this evidence supports the above conclusions.

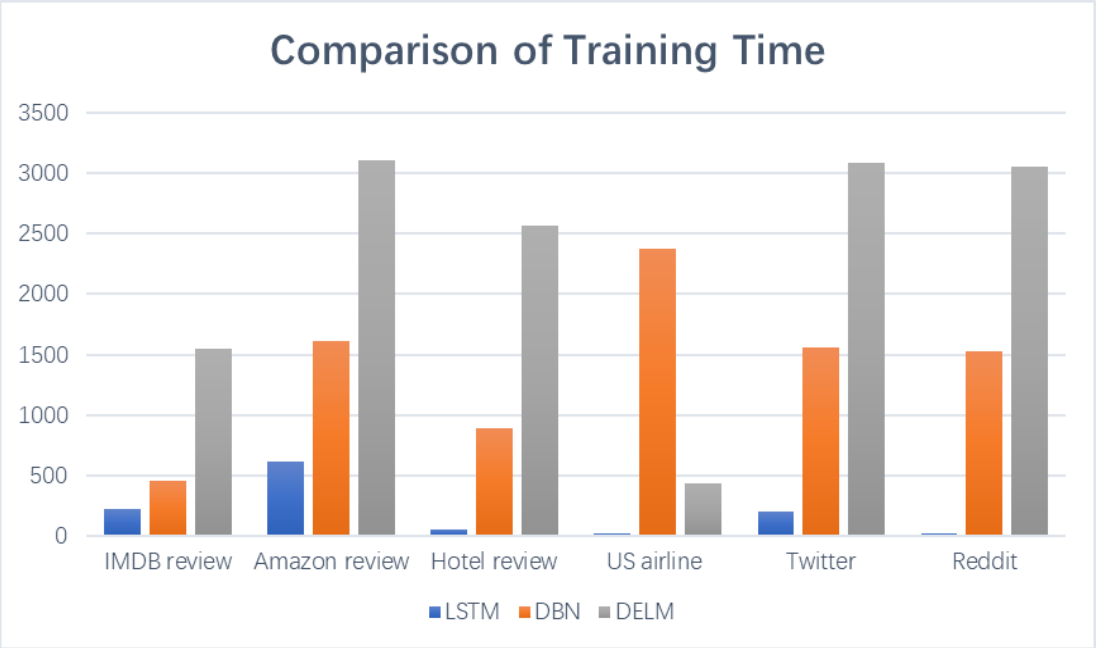

Figure 4-8 Comparison of Training Time

It is difficult to tell which deep neural network obtains the best accuracy in the classification task in Figure 4-9. By looking up the detailed data, LSTM performs the best on most datasets, DBN achieves the best accuracy on the IMDB review dataset, and DELM achieved the best accuracy on the Reddit dataset. In the author's view, the reason for better prediction accuracy when applying LSTM is that the structure of the LSTM model is suitable for dealing with sequential data, and it can get information about the relationship between two close tokens. The reason why DELM and DBN achieve worse prediction accuracy can be explained in this way. DELM and DBN are not designed for sequential data, which makes them miss many useful features in the data. When conducting statistical tests on every pair of data about the accuracy of LSTM, DBN, and DELM models on the same dataset, the significant difference in mean between the accuracy of DBN and DELM on the US airline sentiment achieves the 5% level. Moreover, the significant difference in mean between the accuracy of DBN and DELM on the Twitter dataset achieves the 1% level, and the significant difference in mean between the accuracy of LSTM and DELM on the Reddit dataset

also achieves the 1% level. Thus, LSTM can be considered as achieving the best results on five datasets. However, the significant difference in mean between the other pairs of data never gets the 1% level (refer to tables in Appendix A). Thus, this evidence supports the above conclusions.

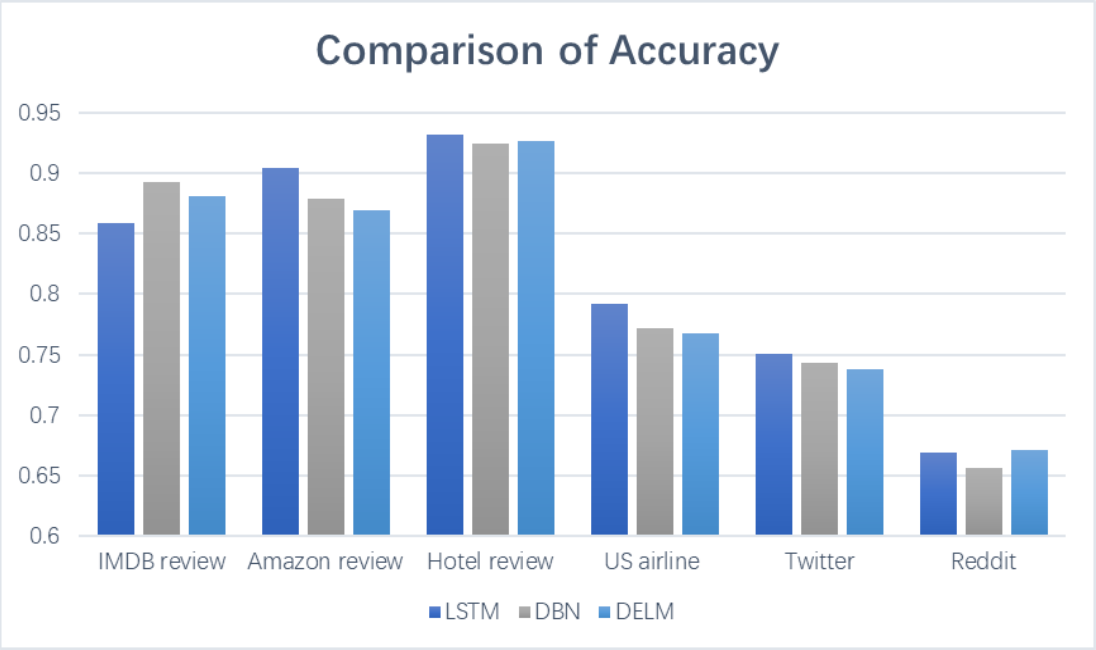

Figure 4-9 Comparison of Accuracy

4.5 SUMMARY

In this chapter, the experimental results show the weakness and strengths of DELM systematically. The author also gives some reasonable inferences to explain the performance of DELM in sentiment analysis. Further, some subtasks are also completed in a series of experiments. First, content statistics are performed on six datasets to determine their suitability for sentiment analysis experiments. Second, multiple data pre-processing approaches are applied to discover a suitable pre-processing way for sentiment analysis on particular datasets.

The primary findings of the experiments and some fact-based inferences are the following:

- The six datasets used in the experiments contain enough useful content, and each has its characteristics.

- Researchers need to pay attention to the details when performing the data pre-processing steps. For example, a series of processing would reset the data of some units to zero, and they need to be deleted.
- All hyper parameters should be adjusted to the optimum values before testing the performance of deep neural networks.
- Keeping the deep neural network unchanged, different datasets would require different training time, depending on the amount of data in the dataset and the hyper parameters in the deep neural network model.
- Keeping the deep neural network unchanged, different datasets would have a different accuracy value. However, under different deep neural networks, the order of accuracy of datasets is consistent. Thus, the accuracy depends mainly on the dataset.
- Keeping the dataset unchanged, different deep neural networks would require different training time. LSTM needs the shortest time, and DELM takes the longest time. However, the gap between the accuracy of LSTM, DBN, and DELM is not big.

Chapter 5: Conclusion and Future Work

This chapter contains conclusions, limitations, and future work.

The chapter begins with a summary paragraph of the research in section 5.1. It is a brief conclusion of everything covered so far. Section 5.2 details what can be improved in this research. Section 5.3 demonstrates the raising of new and pertinent questions for future research.

5.1 CONCLUSIONS

Sentiment analysis is representative of natural language processing (NLP) tasks. It is widely used in society, and its application areas include retail, media, investment, public opinion supervision and so on. A typical sentiment analysis requires four steps: selection of datasets, feature selection, classification, and evaluation of the results. Because deep learning is currently a hot research spot, sentiment analysis is supported by a set of powerful tools, the deep neural networks. Thus, sentiment analysis has evolved toward larger data volumes and higher accuracy.

Deep extreme learning machine (DELM) is a new member of the deep neural networks family. It is developed from extreme learning machine (ELM) which is one of the neural network models that do not use back-propagation. 'Deep' means more hidden layers, stronger computing power, and stronger data processing power. DELM has not been systematically tested for the ability to perform sentiment analysis.

The main purpose of this research is to explore the advantages and disadvantages of DELM in the field of sentiment analysis. To address the purpose, there are four phases in the methodology: (1) Datasets Selection, (2) Text-based Data Pre-processing, (3) Deep Neural Network Classification and (4) Performance Evaluation. In brief, a series of sentiment analysis processes have been done to obtain the experimental results to answer the research question. Several key contributions meet the objectives and sub-objectives of this research:

- Introducing suitable datasets for sentiment classification tasks based on deep neural networks.
- Introducing an effective feature selection approach for text-based sentiment classification tasks based on deep neural networks.
- Demonstrating the strengths and weaknesses of DELM in text-based sentiment classification tasks based on the results of experiments.

The answer to the research question in this research is the following:

After conducting the experiments, the author concludes that DELM currently does not have advantages on accuracy and training time over LSTM and DBN in sentiment analysis. There are two main reasons for the experimental results. First, currently, the word embedding technology is tied to the LSTM model, but it is not

compatible with the DBN model and the DELM model. This restricts the performance of DELM in this research. Second, DELM runs in the MATLAB environment in this research. MATLAB is a memory-intensive program, so the training time of DELM in the experiments is lengthened. However, in the author's opinion, DELM still has the potential to become an effective sentiment classification approach because, in the results, it has close accuracies compared to mainstream approaches under the influence of many unfavorable factors.

5.2 LIMITATIONS

The original intention of this research is to detect the advantages of DELM in sentiment analysis. However, the results demonstrate only few advantages of DELM but many disadvantages. When getting the experimental results like this, the author should conduct further experiments to discover DELM's actual strengths in sentiment analysis. For instance, DELM needs too much training time because of its structure and running in MATLAB. Thus, the author can transfer the codes from MATLAB into Python to make DELM run faster. Furthermore, a more efficient and effective structure for DELM should be developed to reduce the dimensions of the input vectors to reduce runtime.

Moreover, feature selection in this research is not perfect. The features of data in this research only contain information about the presence and frequency of tokens. Some effective feature selection approaches such as Chi-square should be implemented to reduce features. If a better feature selection approaches are adopted, the training time of the classifiers would be greatly reduced, so that there would be one more control group in the experimental results to enhance the persuasiveness of this research.

Finally, attention-based deep neural network models should be applied in this research. In 2018, Google published a novel pre-training model based on deep learning – 'BERT'. It achieved better performance than traditional deep learning models in several NLP tasks (Ning et al., 2019). Sentiment analysis is one of the subtasks in NLP. Thus, the author should compare DELM to 'BERT' to detect the gap.

5.3 FUTURE WORK

Sentiment analysis is important to society in many areas. Deep learning has developed rapidly in recent years. DELM has great potential to be discovered. Therefore, they need more attention in research. Currently, this research has many limitations as mentioned in the previous section. Thus, more work related to this research needs to do in the future.

The author's recommendations are the following:

- To establish a collection of datasets for sentiment analysis.
- To apply more feature selection approaches for raw data in sentiment analysis.
- To identify whether word embedding can be adopted by DBN and DELM models and whether they can be compatible with each other.
- To develop a library out of MATLAB to run DELM easier and faster.
- To try more applications to find the advantages of DELM.
- To identify the gap between DELM and 'BERT', and find the direction of development for DELM.

References

ABBAS, A., ZHOU, Y., DENG, S. & ZHANG, P. 2018. Text analytics to support sense-making in social media: A language-action perspective. *MIS Quarterly,* 42.

ABDEL-ZAHER, A. M. & ELDEIB, A. M. 2016. Breast cancer classification using deep belief networks. *Expert Systems with Applications,* 46**,** 139-144.

AL-AMRANI, Y., LAZAAR, M. & EL KADIRI, K. E. 2018. Sentiment analysis using hybrid method of support vector machine and decision tree. *Journal of Theoretical & Applied Information Technology,* 96.

ALBERTBUP 2017. A Python implementation of Deep Belief Networks built upon NumPy and TensorFlow with scikit-learn compatibility.

ANYANWU, M. N. & SHIVA, S. G. 2009. Comparative analysis of serial decision tree classification algorithms. *International Journal of Computer Science and Security,* 3**,** 230-240.

BABBIE, E. R. 1998. The practice of social research. *Belmont, CA,* 112.

BAO, Y., QUAN, C., WANG, L. & REN, F. The role of pre-processing in twitter sentiment analysis. International Conference on Intelligent Computing, 2014. Springer, 615-624.

BERNARD, H. R. 2013. *Social research methods: Qualitative and quantitative approaches*, Sage.

BOLLEN, J., MAO, H. & ZENG, X. 2011. Twitter mood predicts the stock market. *Journal of computational science,* 2**,** 1-8.

BOUMA, G. 2009. Normalized (pointwise) mutual information in collocation extraction. *Proceedings of GSCL***,** 31-40.

CHOLLET, F. 2015. Keras.

DE ALBORNOZ, J. C., PLAZA, L., GERVÁS, P. & DÍAZ, A. A joint model of feature mining and sentiment analysis for product review rating. European conference on information retrieval, 2011. Springer, 55-66.

DEDINEC, A., FILIPOSKA, S., DEDINEC, A. & KOCAREV, L. 2016. Deep belief network based electricity load forecasting: An analysis of Macedonian case. *Energy,* 115**,** 1688-1700.

FERNÁNDEZ, A., GARCÍA, S., DEL JESUS, M. J. & HERRERA, F. 2008. A study of the behaviour of linguistic fuzzy rule based classification systems in the framework of imbalanced data-sets. *Fuzzy Sets and Systems,* 159**,** 2378-2398.

GHASEMI, A. & ZAHEDIASL, S. 2012. Normality tests for statistical analysis: a guide for non-statisticians. *International journal of endocrinology and metabolism,* 10**,** 486.

GLOROT, X., BORDES, A. & BENGIO, Y. Domain adaptation for large-scale sentiment classification: A deep learning approach. Proceedings of the 28th international conference on machine learning (ICML-11), 2011. 513-520.

GODBOLE, N., SRINIVASAIAH, M. & SKIENA, S. 2007. Large-Scale Sentiment Analysis for News and Blogs. *Icwsm,* 7**,** 219-222.

GOODFELLOW, I., BENGIO, Y. & COURVILLE, A. 2016. Sequence Modeling: Recurrent and Recursive Nets. *Deep Learning***,** 367-415.

GU, Y., CHEN, Y., LIU, J. & JIANG, X. 2015. Semi-supervised deep extreme learning machine for Wi-Fi based localization. *Neurocomputing,* 166**,** 282-293.

HASTIE, T., TIBSHIRANI, R. & FRIEDMAN, J. 2009. Unsupervised learning. *The elements of statistical learning.* Springer.
HINTON, G. E., OSINDERO, S. & TEH, Y.-W. 2006. A fast learning algorithm for deep belief nets. *Neural computation,* 18**,** 1527-1554.
HINTON, G. E., SRIVASTAVA, N., KRIZHEVSKY, A., SUTSKEVER, I. & SALAKHUTDINOV, R. R. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.
HUANG, G.-B., ZHU, Q.-Y. & SIEW, C.-K. 2006. Extreme learning machine: theory and applications. *Neurocomputing,* 70**,** 489-501.
HUANG, Y. & BIAN, L. 2009. A Bayesian network and analytic hierarchy process based personalized recommendations for tourist attractions over the Internet. *Expert Systems with Applications,* 36**,** 933-943.
KIRITCHENKO, S., ZHU, X. & MOHAMMAD, S. M. 2014. Sentiment analysis of short informal texts. *Journal of Artificial Intelligence Research,* 50**,** 723-762.
LECUN, Y., BENGIO, Y. & HINTON, G. 2015. Deep learning. *nature,* 521**,** 436.
LI, X., XIE, H., WANG, R., CAI, Y., CAO, J., WANG, F., MIN, H. & DENG, X. 2016. Empirical analysis: stock market prediction via extreme learning machine. *Neural Computing and Applications,* 27**,** 67-78.
LIU, B. 2015. *Sentiment analysis: Mining opinions, sentiments, and emotions*, Cambridge University Press.
MA, X., TAO, Z., WANG, Y., YU, H. & WANG, Y. 2015. Long short-term memory neural network for traffic speed prediction using remote microwave sensor data. *Transportation Research Part C: Emerging Technologies,* 54**,** 187-197.
MAKNICKIENE, N., LAPINSKAITE, I. & MAKNICKAS, A. 2018. Application of ensemble of recurrent neural networks for forecasting of stock market sentiments. *Equilibrium. Quarterly Journal of Economics and Economic Policy,* 13**,** 7-27.
MÄNTYLÄ, M. V., GRAZIOTIN, D. & KUUTILA, M. 2018. The evolution of sentiment analysis—A review of research topics, venues, and top cited papers. *Computer Science Review,* 27**,** 16-32.
MEDHAT, W., HASSAN, A. & KORASHY, H. 2014. Sentiment analysis algorithms and applications: A survey. *Ain Shams engineering journal,* 5**,** 1093-1113.
MIKOLOV, T., SUTSKEVER, I., CHEN, K., CORRADO, G. S. & DEAN, J. Distributed representations of words and phrases and their compositionality. Advances in neural information processing systems, 2013. 3111-3119.
MOHAMMAD, S., DUNNE, C. & DORR, B. Generating high-coverage semantic orientation lexicons from overtly marked words and a thesaurus. Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2, 2009. Association for Computational Linguistics, 599-608.
NING, Z., LIN, Y. & ZHONG, R. Team Peter-Parker at SemEval-2019 Task 4: BERT-Based Method in Hyperpartisan News Detection. Proceedings of the 13th International Workshop on Semantic Evaluation, 2019. 1037-1040.
O'HARE, N., DAVY, M., BERMINGHAM, A., FERGUSON, P., SHERIDAN, P., GURRIN, C. & SMEATON, A. F. Topic-dependent sentiment analysis of financial blogs. Proceedings of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion, 2009. ACM, 9-16.
PAK, A. & PAROUBEK, P. Twitter as a corpus for sentiment analysis and opinion mining. LREc, 2010. 1320-1326.

PANG, B. & LEE, L. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. Proceedings of the 42nd annual meeting on Association for Computational Linguistics, 2004. Association for Computational Linguistics, 271.
PANG, B. & LEE, L. 2008. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval,* 2**,** 1-135.
PARLAR, T., ÖZEL, S. A. & SONG, F. 2018. QER: a new feature selection method for sentiment analysis. *Human-centric Computing and Information Sciences,* 8**,** 10.
PORIA, S., CAMBRIA, E., HOWARD, N., HUANG, G.-B. & HUSSAIN, A. 2016. Fusing audio, visual and textual clues for sentiment analysis from multimodal content. *Neurocomputing,* 174**,** 50-59.
PRISS, U. 2006. Formal concept analysis in information science. *Annual review of information science and technology,* 40**,** 521-543.
QAZI, A., RAJ, R. G., HARDAKER, G. & STANDING, C. 2017. A systematic literature review on opinion types and sentiment analysis techniques: Tasks and challenges. *Internet Research,* 27**,** 608-630.
RAVI, K. & RAVI, V. 2015. A survey on opinion mining and sentiment analysis: tasks, approaches and applications. *Knowledge-Based Systems,* 89**,** 14-46.
ROIGER, R. J. 2017. *Data mining: a tutorial-based primer*, Chapman and Hall/CRC.
ROUL, R. K., ASTHANA, S. R. & KUMAR, G. 2017. Study on suitability and importance of multilayer extreme learning machine for classification of text data. *Soft Computing,* 21**,** 4239-4256.
RUANGKANOKMAS, P., ACHALAKUL, T. & AKKARAJITSAKUL, K. Deep belief networks with feature selection for sentiment classification. 2016 7th International Conference on Intelligent Systems, Modelling and Simulation (ISMS), 2016. IEEE, 9-14.
SAINATH, T. N., VINYALS, O., SENIOR, A. & SAK, H. Convolutional, long short-term memory, fully connected deep neural networks. 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2015. IEEE, 4580-4584.
SCHMIDHUBER, J. 2015. Deep learning in neural networks: An overview. *Neural networks,* 61**,** 85-117.
SERRANO-GUERRERO, J., OLIVAS, J. A., ROMERO, F. P. & HERRERA-VIEDMA, E. 2015. Sentiment analysis: A review and comparative analysis of web services. *Information Sciences,* 311**,** 18-38.
SOCHER, R., PERELYGIN, A., WU, J., CHUANG, J., MANNING, C. D., NG, A. & POTTS, C. Recursive deep models for semantic compositionality over a sentiment treebank. Proceedings of the 2013 conference on empirical methods in natural language processing, 2013. 1631-1642.
SONG, W. & PARK, S. C. 2009. Genetic algorithm for text clustering based on latent semantic indexing. *Computers & Mathematics with Applications,* 57**,** 1901-1907.
SUN, K., ZHANG, J., ZHANG, C. & HU, J. 2017. Generalized extreme learning machine autoencoder and a new deep neural network. *Neurocomputing,* 230**,** 374-381.
TABOADA, M., BROOKE, J., TOFILOSKI, M., VOLL, K. & STEDE, M. 2011. Lexicon-based methods for sentiment analysis. *Computational linguistics,* 37**,** 267-307.

TANG, D., QIN, B. & LIU, T. Document modeling with gated recurrent neural network for sentiment classification. Proceedings of the 2015 conference on empirical methods in natural language processing, 2015a. 1422-1432.
TANG, J., DENG, C. & HUANG, G.-B. 2015b. Extreme learning machine for multilayer perceptron. *IEEE transactions on neural networks and learning systems,* 27**,** 809-821.
THASEEN, I. S. & KUMAR, C. A. 2017. Intrusion detection model using fusion of chi-square feature selection and multi class SVM. *Journal of King Saud University-Computer and Information Sciences,* 29**,** 462-472.
THELWALL, M. & BUCKLEY, K. 2013. Topic‐based sentiment analysis for the social web: The role of mood and issue‐related words. *Journal of the American Society for Information Science and Technology,* 64**,** 1608-1617.
TING, S., IP, W. & TSANG, A. H. 2011. Is Naive Bayes a good classifier for document classification. *International Journal of Software Engineering and Its Applications,* 5**,** 37-46.
VARIOR, R. R., SHUAI, B., LU, J., XU, D. & WANG, G. A siamese long short-term memory architecture for human re-identification. European conference on computer vision, 2016. Springer, 135-153.
WAN, Y. & GAO, Q. An ensemble sentiment classification system of twitter data for airline services analysis. 2015 IEEE international conference on data mining workshop (ICDMW), 2015. IEEE, 1318-1325.
WANG, H., WANG, G., LI, G., PENG, J. & LIU, Y. 2016a. Deep belief network based deterministic and probabilistic wind speed forecasting approach. *Applied Energy,* 182**,** 80-93.
WANG, Y., HUANG, M. & ZHAO, L. Attention-based LSTM for aspect-level sentiment classification. Proceedings of the 2016 conference on empirical methods in natural language processing, 2016b. 606-615.
WANG, Z. & PARTH, Y. 2016. Extreme learning machine for multi-class sentiment classification of tweets. *Proceedings of ELM-2015 Volume 1.* Springer.
YAMADA, K. D. & KINOSHITA, K. 2018. De novo profile generation based on sequence context specificity with the long short-term memory network. *BMC bioinformatics,* 19**,** 272.
YU, H. & KIM, S. 2012. SVM tutorial—classification, regression and ranking. *Handbook of Natural computing***,** 479-506.
YU, W., ZHUANG, F., HE, Q. & SHI, Z. 2015. Learning deep representations via extreme learning machines. *Neurocomputing,* 149**,** 308-315.
ZHANG, J., DING, S., ZHANG, N. & SHI, Z. 2016a. Incremental extreme learning machine based on deep feature embedded. *International journal of machine learning and cybernetics,* 7**,** 111-120.
ZHANG, K., CHAO, W.-L., SHA, F. & GRAUMAN, K. Video summarization with long short-term memory. European conference on computer vision, 2016b. Springer, 766-782.
ZHAO, Z., JIAO, L., ZHAO, J., GU, J. & ZHAO, J. 2017. Discriminant deep belief network for high-resolution SAR image classification. *Pattern Recognition,* 61**,** 686-701.
ZIEBART, B. D., MAAS, A. L., BAGNELL, J. A. & DEY, A. K. Maximum entropy inverse reinforcement learning. Aaai, 2008. Chicago, IL, USA, 1433-1438.

Appendices

Appendix A

Statistical Test Outcomes

Table A 1 Dataset Comparison Based on LSTM Training Time (P-value)

Dataset	IMDB review	Amazon review	Hotel review	US airline sentiment	Twitter
Amazon review	8.83×10^{-5}				
Hotel review	8.39×10^{-5}	4.10×10^{-34}			
US airline sentiment	8.54×10^{-5}	1.49×10^{-34}	1.64×10^{-33}		
Twitter	7.04×10^{-5}	7.29×10^{-5}	6.91×10^{-5}	7.04×10^{-5}	
Reddit	8.54×10^{-5}	1.26×10^{-34}	4.69×10^{-36}	2.28×10^{-20}	7.04×10^{-5}

Table A 2 Dataset Comparison Based on LSTM Accuracy (P-value)

Dataset	IMDB review	Amazon review	Hotel review	US airline sentiment	Twitter
Amazon review	6.97×10^{-9}				
Hotel review	1.75×10^{-12}	5.14×10^{-14}			
US airline sentiment	8.83×10^{-5}	8.83×10^{-5}	8.88×10^{-5}		
Twitter	1.60×10^{-15}	4.66×10^{-27}	2.00×10^{-33}	8.83×10^{-5}	
Reddit	1.82×10^{-19}	1.08×10^{-27}	2.55×10^{-30}	8.88×10^{-5}	5.91×10^{-21}

Table A 3 Dataset Comparison Based on DBN Training Time (P-value)

Dataset	IMDB review	Amazon review	Hotel review	US airline sentiment	Twitter
Amazon review	9.08×10^{-5}				
Hotel review	7.81×10^{-28}	9.03×10^{-5}			
US airline sentiment	9.08×10^{-5}	9.13×10^{-5}	9.03×10^{-5}		
Twitter	3.05×10^{-33}	5.02×10^{-4}	7.09×10^{-29}	9.08×10^{-5}	
Reddit	4.51×10^{-29}	9.08×10^{-5}	9.31×10^{-25}	9.08×10^{-5}	1.29×10^{-4}

Table A 4 Dataset Comparison Based on DBN Accuracy (P-value)

Dataset	IMDB review	Amazon review	Hotel review	US airline sentiment	Twitter
Amazon review	1.49×10^{-6}				
Hotel review	1.85×10^{-15}	1.60×10^{-16}			
US airline sentiment	5.49×10^{-17}	7.83×10^{-16}	3.56×10^{-19}		
Twitter	8.98×10^{-5}	9.03×10^{-5}	9.03×10^{-5}	9.03×10^{-5}	
Reddit	3.33×10^{-26}	3.91×10^{-25}	2.87×10^{-28}	4.74×10^{-16}	9.08×10^{-5}

Table A 5 Dataset Comparison Based on DELM Training Time (P-value)

Dataset	IMDB review	Amazon review	Hotel review	US airline sentiment	Twitter
Amazon review	9.13×10^{-5}				
Hotel review	9.13×10^{-5}	9.13×10^{-5}			
US airline sentiment	9.08×10^{-5}	9.08×10^{-5}	9.08×10^{-5}		
Twitter	9.13×10^{-5}	$\mathbf{0.55}$	9.13×10^{-5}	9.08×10^{-5}	
Reddit	9.13×10^{-5}	$\mathbf{0.09}$	9.13×10^{-5}	9.08×10^{-5}	$\mathbf{0.41}$

Table A 6 Dataset Comparison Based on DELM Accuracy (P-value)

Dataset	IMDB review	Amazon review	Hotel review	US airline sentiment	Twitter
Amazon review	1.34×10^{-8}				
Hotel review	2.53×10^{-21}	2.61×10^{-21}			
US airline sentiment	5.25×10^{-20}	9.06×10^{-19}	4.92×10^{-23}		
Twitter	1.15×10^{-27}	9.40×10^{-26}	3.86×10^{-32}	8.66×10^{-10}	
Reddit	3.11×10^{-32}	4.11×10^{-30}	5.25×10^{-38}	5.03×10^{-19}	3.52×10^{-23}

Table A 7 Algorithm Comparison for Given Datasets Based on Training Time (P-value)

Datasets		LSTM	DBN
IMDB review	**DELM**	8.83×10^{-5}	9.08×10^{-5}
	LSTM		8.78×10^{-5}
Amazon review	**DELM**	1.12×10^{-28}	9.13×10^{-5}
	LSTM		9.13×10^{-5}
Hotel review	**DELM**	8.68×10^{-5}	9.03×10^{-5}
	LSTM		6.38×10^{-35}
US airline sentiment	**DELM**	8.78×10^{-5}	9.08×10^{-5}
	LSTM		8.83×10^{-5}
Twitter	**DELM**	7.29×10^{-5}	4.15×10^{-20}
	LSTM		7.25×10^{-5}
Reddit	**DELM**	5.93×10^{-27}	2.31×10^{-21}
	LSTM		4.01×10^{-32}

Table A 8 Algorithm Comparison for Given Datasets Based on Accuracy (P-value)

Datasets		LSTM	DBN
IMDB review	**DELM**	7.08×10^{-5}	7.41×10^{-7}
	LSTM		6.35×10^{-7}
Amazon review	**DELM**	1.96×10^{-14}	3.94×10^{-5}
	LSTM		1.38×10^{-10}
Hotel review	**DELM**	3.87×10^{-8}	5.35×10^{-4}
	LSTM		4.35×10^{-9}
US airline sentiment	**DELM**	5.61×10^{-4}	$\mathbf{0.36}$
	LSTM		1.38×10^{-3}
Twitter	**DELM**	6.99×10^{-10}	$\mathbf{1.28 \times 10^{-2}}$
	LSTM		3.81×10^{-4}
Reddit	**DELM**	$\mathbf{0.22}$	2.59×10^{-6}
	LSTM		1.51×10^{-4}