

**DEEP LEARNING MODEL FOR OPINION MINING**

**BY**

**LEE HAO JIE**

**A REPORT**

**SUBMITTED TO**

**Universiti Tunku Abdul Rahman**

**in partial fulfillment of the requirements**

**for the degree of**

**BACHELOR OF COMPUTER SCIENCE (HONOURS)**

**Faculty of Information and Communication Technology**

**(Kampar Campus)**

**Oct 2022**

## REPORT STATUS DECLARATION FORM

**Title:** \_\_\_\_\_ DEEP LEARNING MODEL FOR OPINION MINING \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

**Academic Session:** \_\_\_2022 Oct\_\_\_

I \_\_\_\_\_ Lee Hao Jie \_\_\_\_\_

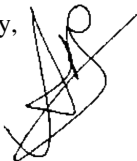
(CAPITAL LETTER)

declare that I allow this Final Year Project Report to be kept in  
Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.



\_\_\_\_\_  
(Author's signature)

Verified by, 

\_\_\_\_\_  
(Supervisor's signature)

**Address:**

\_\_No99, Taman Kagangan,\_\_\_\_\_  
\_\_36400, Hutan Melintang,\_\_\_\_\_  
\_\_Perak\_\_\_\_\_

Dr.Ramesh Kumar Ayyasamy  
\_\_\_\_\_

Supervisor's name

**Date:** \_\_\_29/11/2022\_\_\_\_\_

**Date:** \_\_\_30/11/2022\_\_\_\_\_

<b>Universiti Tunku Abdul Rahman</b>			
Form Title: <b>Sample of Submission Sheet for FYP/Dissertation/Thesis</b>			
Form Number: <b>FM-IAD-004</b>	Rev No.: <b>0</b>	Effective Date: <b>21 JUNE 2011</b>	Page No.: <b>1 of 1</b>

**FACULTY/INSTITUTE\* OF Information and Communication Technology**

**UNIVERSITI TUNKU ABDUL RAHMAN**

Date: 29/11/2022

**SUBMISSION OF FINAL YEAR PROJECT /DISSERTATION/THESIS**

It is hereby certified that Lee Hao Jie (ID No: 18ACB01544) has completed this final year project entitled “Deep Learning Model for Opinion Mining” under the supervision of Dr Ramesh Kumar Ayyasamy (Supervisor) from the Department of Computer Science, Faculty/Institute\* of Information and Communication Technology.

I understand that University will upload softcopy of my final year project in pdf format into UTAR Institutional Repository, which may be made accessible to UTAR community and public.

Yours truly,




\_\_\_\_\_  
(Lee Hao Jie)

\*Delete whichever not applicable

## DECLARATION OF ORIGINALITY

I declare that this report entitled “**DEEP LEARNING MODEL FOR OPINION MINING**” is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature :  \_\_\_\_\_

Name : Lee Hao Jie \_\_\_\_\_

Date : 29/11/2022 \_\_\_\_\_

## **ACKNOWLEDGEMENTS**

I would like to extend my sincere thanks to my supervisor Dr Ramesh Kumar Ayyasamy. He allows me to explore this exciting title. Dr Ramesh also guides me when I lose direction. This is an opportunity to start the journey in deep learning.

I 'm incredibly grateful to my family, who support me on my way to college. In this project, they provided support and financial support to buy me a desktop for me could do this project better. I will use the result to appreciate the help they give to me.

## **ABSTRACT**

LSTM (Long Short-Term Memory) shows its performance in Sentiment Analysis, but it has a critical drawback in terms of how to do backpropagation, limiting the training time to more extended and the process slower. Attention mechanism more behavior like human understands the sentences by a focus on specific words to solve the issue from LSTM. The Bert (Bidirectional Encoder Representations from Transformers) use an attention mechanism and outperform other attention-based model such as GPT (Generative Pre-trained Transformer) and Elmo (Embeddings from Language Model) because it has learned the deep bidirectional presentations by the MLM (Masked Language Model) and NSP (Next Sentence Prediction).

Ernie (Enhanced Language Representation with Informative Entities) model and Zen model modify how Bert model learns language and gains achievement in Chinese NLP (Natural Language Process). RoBERTa (A Robustly Optimized BERT Pretraining Approach) from Facebook proves the NSP is not helping the model, so we also modify the NSP task others for to learn the language. Sentiment Analysis is one of the NLP tasks Ernie and Zen successful in beating the Bert-Chinese, which uses Chinese characters as input and WordPiece embeddings to do word embeddings. Word level embeddings and input is needed to improve the Bert model works on Chinese Sentiment Analysis.

With the motivation to improve the Chinese Sentiment Analysis, this project will combine experience from different models to propose a better version of the Bert model. This project will limit the scope to improve Sentiment Analysis among different NLP tasks.

# TABLE OF CONTENTS

<b>TITLE PAGE</b>	<b>i</b>
<b>REPORT STATUS DECLARATION FORM</b>	<b>ii</b>
<b>FYP THESIS SUBMISSION FORM</b>	<b>iii</b>
<b>DECLARATION OF ORIGINALITY</b>	<b>iv</b>
<b>ACKNOWLEDGEMENTS</b>	<b>v</b>
<b>ABSTRACT</b>	<b>vi</b>
<b>TABLE OF CONTENTS</b>	<b>vii</b>
<b>LIST OF FIGURES</b>	<b>x</b>
<b>LIST OF ABBREVIATIONS</b>	<b>xi</b>
<b>CHAPTER 1 Introduction</b>	<b>1</b>
1.1 Problem Statement and Motivation	1
1.2 Objectives	2
1.3 Project Scope and Direction	3
1.4 Contributions	4
1.5 Background Information	5
1.5.1 Definition of sentimental analysis and opinion	5
1.5.2 Level of sentiment analysis	6
1.5.3 Process of sentimental analysis	6
1.5.4 Introduction to classification method	7
1.6 Report Organization	7
<b>CHAPTER 2 Literature Review</b>	<b>8</b>
2.1 Sentimental analysis in general	8
2.1.1 Feature selection on sentimental analysis	8
2.1.2 Technique on sentimental analysis	9
2.1.3 Deep Learning in Sentimental Analysis	11
2.2 Chinese text sentimental analysis with deep learning approach	11
2.2.1 Convolutional neural network	12
2.2.2 Long Short-Term Memory	13

2.2.3	BERT	15
<b>CHAPTER 3 System Model</b>		<b>17</b>
3.1	Overview of the Model and development process	17
3.2	Pre-train Model explain	18
3.3	Model framework and framework version	19
3.4	Dataset	19
3.5	Model Design	20
3.5.1	Encoder	21
3.5.2	Attention Layer	22
3.5.3	Position Wise Feed Forward Layer	23
<b>CHAPTER 4 Data per-process and Model output</b>		<b>24</b>
4.1	Tokenization Process	24
4.2	Model Input	26
4.3	Model output	27
<b>CHAPTER 5 EXPERIMENT</b>		<b>28</b>
5.1	Environment Setup	28
5.2	Train Parameters	28
5.3	Fine-Tune Process	29
5.4	Infer with one Sample	30
5.5	Implementation Issues and Challenges	31
<b>CHAPTER 6 SYSTEM EVALUATION AND DISCUSSION</b>		<b>32</b>
6.1	Testing Setup	32
6.2	Model Accuracy Metrics	32
6.3	Objectives Evaluation and Concluding Remark	36
6.4	Future work and challenges	36



<b>CHAPTER 7 Conclusion and Recommendation</b>	<b>37</b>
7.1 Conclusion	37
7.2 Recommendation	37
<b>REFERENCES</b>	<b>A-1</b>
<b>WEEKLY LOG</b>	<b>A-5</b>
<b>POSTER</b>	<b>A-11</b>
<b>PLAGIARISM CHECK RESULT</b>	<b>A-12</b>
<b>FYP2 CHECKLIST</b>	<b>A-18</b>

## LIST OF FIGURES

<b>Figure Number</b>	<b>Title</b>	<b>Page</b>
Figure 2.1.1	Sentimental Analysis Techniques	9
Figure 3.1	Model Development Process	17
Figure 3.2	Pretrain process	18
Figure 3.4	Records of the ChnSentiCorp	19
Figure 3.5.1	Ernie Model in general	20
Figure 3.5.2	Ernie Model input and output	20
Figure 3.5.3	An Encoder Block	21
Figure 3.5.4	An Attention Layer	22
Figure 3.5.5	A Point Wise Feed Forward Layer	23
Figure 4.1.1	Vocab.txt	24
Figure 4.1.2	Tokenization Process	25
Figure 4.1.3	Resource Map	25
Figure 4.1.4	Regex pattern for Tokenizer	26
Figure 4.2.1	Model input in general view	26
Figure 4.2.2	Model input in actual view	27
Figure 4.3.1	System Architecture Diagram	27
Figure 5.3.1	Fine Tune Process	29
Figure 5.3.2	Fine Tune Process2	29
Figure 5.4.1	Sample Record	30
Figure 5.4.2	Tokenization output and reshape of dimension	30
Figure 5.4.3	Output of the model	31
Figure 6.2.1	Binary Cross Entropy graphs	33
Figure 6.2.2	Loss based on Epochs line graph	33
Figure 6.2.3	Accuracy based on Epochs line graph	34
Figure 6.2.4	F1-score based on Epochs line graph	35

## LIST OF ABBREVIATIONS

<i>Bert</i>	Bidirectional Encoder Representations from Transformers
<i>CNN</i>	Convolution Neural Network
<i>DLM</i>	Dialogue Language Model
<i>Elmo</i>	Embeddings from Language Model
<i>Ernie</i>	Enhanced Language Representation with Informative Entities
<i>et al.</i>	and others
<i>GBK</i>	Chinese Internal Code Extension Specification
<i>GPT</i>	Generative Pre-trained Transformer
<i>GPU</i>	Graphic Processing Unit
<i>ids</i>	Indexes
<i>LSTM</i>	Long Short-Term Memory
<i>MLM</i>	Masked Language Model
<i>NLP</i>	Natural Language Processing
<i>NSP</i>	Next Sentence Prediction
<i>PAD</i>	Padding
<i>POS</i>	Part-of-Speech
<i>ReLU</i>	Rectified Linear Units
<i>RoBERTa</i>	A Robustly Optimized BERT Pretraining Approach
<i>UTF</i>	Unicode Transformation Format

# Chapter 1

## Introduction

### 1.1 Problem Statement and Motivation

The massive growth of the World Wide Web brings convenience to people, especially the ease of buying a product online. The existence of the e-commerce platform provides a platform for the user to share their feeling, thought, or opinion on the product. Reviews from others can influence others' decisions. People will refer to other individuals' opinions to compare and judge a thing or a product. For instance, most customers will analyze others' reviews to decide before buying the product. Review on the internet is an important channel for the company to gain feedback from the customer. In addition, the company wants to build up its credibility, so they gather opinions from users to improve its product. In nature, the analysis process of the reviews from the e-commerce platform is tedious and time-consuming. With the requirement to analyze the review, Sentimental Analysis has become a new research area, sentimental analysis targets abstracting the opinion and sentiment from the text. With Sentimental Analysis, the company can process many reviews or comments to get important information. Sentimental Analysis is also known as opening mining. The sentimental analysis output from a word, sentence, or paragraph is either positive, negative, or neutral. The company can apply Sentimental Analysis to analyze a review's emotional context and quickly classify different reviews into two categories. Some people will question whether we can do the Sentimental Analysis task with a model to search the review's pre-defined emotional good and bad words. Sentimental Analysis is not that easy because a language is relatively complex for a machine to understand its context. An opinion can be expressed indirectly, such as sarcasm, and the term polarity may change in different domains.

Sentimental Analysis has been applied in different areas, from reviews Analysis to healthcare materials analysis in physiology. Analysis task always is very tedious and repetitive. Recently machine learning has become popular because it is easy to imply and produce a good accuracy model in the classification and regression tasks. In addition, the rising of GPU brings machine learning to another level because of parallel

processing. The neural network was invented by researchers by the conception of the human brain to process information. With GPU, these neural networks can be trained and simulate signals like the human brain processes information.

Moreover, deep learning models such as CNN achieve good accuracy in Image processing tasks. On the other hand, many researchers experimented with different deep learning models in the Natural language Processing task to improve the NLP task. Recently Devlin [3] found out the importance of the attention-based model from the transformer model and developed the BERT model. It outperforms many models such as LSTM in the NLP task, including the Sentimental Analysis. One of the problems of the LSTM model is it cannot handle long sentences or documents. The performance will drop after the input token is more than 500.

While sentimental Analysis has reached a milestone in the English language, the NLP task is still blank in other languages. Chinese is one of the languages with many native speakers in the world. There is a requirement for Chinese sentimental Analysis. To improve and automate the Chinese text analysis task, we will propose a Chinese text sentimental analysis model with a deep learning approach.

## **1.2 Objectives**

There are three sentimental analysis levels: document level, sentence level, and feature level. Document-level sentiment analysis extracts the sentiment value from the full review, while sentence-level sentiment analysis extracts sentiment value from one sentence. Feature-level sentiment analysis first determines the feature or object in the sentences, then classifies the sentiment to determine the opinion on that feature. This project's scope is limited to document-level sentiment analysis, with the rising e-commerce platforms in China such as Taobao. Chinese sentimental analysis has become a requirement for those companies to analyze the review. Chinese has become one of the languages with many native speakers in the world. Most of the sentiment analysis models and the resources are made for English language speakers, so that this project will limit the scope of the language model to Chinese. Sentimental analysis is a classification task.

The Bert model recently achieved state-of-the-art performance in NLP tasks. The Bert model was trained in many languages, including Chinese. Liu [5] teams from Facebook and Washington University question the Bert model on Next Sentences Prediction importance and prove that removing it will outperform the original Bert model when fine-tuning. In addition, Bert on the Chinese tokenizer that uses the WordPiece embeddings forces the model to take character-level input. Random masking technique from the Bert model on character level input could make the Bert model learn nothing but character level context meaning. The issues can be improved by applying the different levels of masking. This project will modify the original Bert model to improve its performance by changing the strategy to pre-train the model and apply different masking levels. We will use PaddlePaddle as our deep learning framework. PaddlePaddle is an open-source machine learning framework that works in from Baidu. In short, this project will be doing document-level sentiment analysis on Chinese text by changing the Bert model to propose a better Chinese sentiment analysis model compared to other models such as CNN and LSTM. The final objective is to find the optimal parameters for the deep learning model to do opinion mining.

### **1.3 Project Scope and Direction**

This research aims to find the best deep learning model to do Chinese sentimental analysis. The project's primary goal is to propose a framework that can gain higher accuracy in Chinese sentimental analysis. The direction of this project is using a pre-train language model and make it specialize on Chinese sentiment analysis.

Bert's model may work well in the English language but not work in the Chinese language. English is easier to encode in embedding space, and words are separated by space characters, unlike Chinese words, which do not exist separate by the area. Bert Chinese model work on character level encoding is not enough to make the Bert model adapt to Chinese NLP tasks. The Bert Model from Zenz and ERNIE models are the current state of art pre-training language models. These models train on multiple NLP tasks, including Named entity recognition, sentiment analysis, and retrieval question answering.

The project only scopes to improve Sentiment analysis accuracy in the Chinese language. This project will experiment on fine tune a pre-training model with Chinese word-level embedding to produce a better Chinese sentimental analysis model. Bert Model applies random masked strategy on Chinese character can be improve by applies different level of masking such as word-level, character level and entity level. The pre-train model Ernie [12] has been pretrain with different level of masking which has better understanding of context Chinese language, so Ernie will be used to fine tune it with ChnSentiCorp to make the model specialize on Chinese sentimental analysis.

A deep learning model is easy to overfit. The overfit deep learning model may outperform other models on the specific dataset but fall short when using other datasets as testing data. To achieve the objective, we will apply dropout layer and sperate dataset to train set and test set to avoid overfit problem. For comparison purposes, this project will use ChnSentiCorp as the dataset to train and evaluate the result in terms of accuracy. ChnSentiCorp is a dataset from hotel review data. It is a well-known dataset and acts as a benchmark for Sentimental Chinese analysis. The system will run on an online Jupyter notebook which is the Baidu Ai Studio. This environment set up on this online Jupyter notebook is easier compared to set up in local computer and it provide user free graphic processing unit.

#### **1.4 Contributions**

Chinese Sentimental Analysis did not get attention from the public for its application. One of the reasons is that the model is not mature enough to deploy for the application. For the English language, Sentimental analysis has been applied to study the feature of their new product. Instead of doing a new survey to gather people's opinions, Sentimental analysis can be used to extract information from the existing resource on the Web. In general, sentimental analysis make the machine can understand the sentimental from the text. Sentiment Analysis can be applied to any task that has the requirement to classify and summarize the sentimental value. To produce a model that framework suitable to deploy, it must meet some specifications such as high accuracy, less time to train, or can use fine-tuning methods to fit in a specific domain. This project will provide a better approach to Chinese sentimental analysis, which can work among different datasets.

## 1.5 Background information

### 1.5.1 Definition of sentimental analysis and opinion

Sentiment analysis is the Natural Language Processing task where we use a model to classify the sentimental polarity of sentences or documents. People may confuse the meaning of Opinion mining and sentimental analysis. According to Medhat [7], sentimental analysis is the subset of opinion mining. Opinion mining is the text mining process with computational linguistics techniques to extract information. The difference between these two processes is opinion mining wants to get more information, such as who and which part has the opinion, and it can be subjective or objective. The sentimental analysis classifies personal documents into polarities such as positive or negative, and sometimes neutral polarity is also included. Therefore, sentiment analysis can be considered an opinion mining task, classifying opinion into different polarities.

According to Liu Bing [5], an opinion can be expressed with or without the object. Features in sentences can be explicit or implicit, different orientations either indirectly or comparatively, and could be subjective or objective. In deep learning, the feature is the model's variable, but the feature in Sentiment analysis could refer to the object feature. For example, in "The battery life of the phone," the phone is the object, and battery life is the feature. One type of sentiment analysis needs to determine the feature, which is feature-based sentiment analysis. This project is non-feature-based sentiment analysis, so to avoid confusion, the term "feature" indicates the model's variable. Second, explicit expression is a direct expression in contrast to implicit expression, which does not indicate the object or the feature. For instance, "The phone is too expensive but has good camera feature." the price and camera are the phone's features. From the example, price is implicit, and the camera is explicit because the word "Price" is not mentioned in the sentences.

Regarding the opinionated document, a sentence's orientation also could affect how the machine gets the sentimental context. One type of orientation is the direct opinion which has an adjective or opinion word in the document. One the other way row is a comparative opinion which compares two features to show similarities or



differences. With the direct opinion orientation, one of the easier ways to do sentimental analysis is to find or build the lexicon or corpus, which acts like a dictionary for the model to get the sentimental value. There are also two types of sentences which is subjective and objective. The objective document is not sentimental because it expresses those personal feelings do not influence it. In conclusion, an opinionated sentence is a sentence that says a good or bad opinion explicitly or implicitly, and it can be a subjective or objective sentence [5].

### **1.5.2 Level of sentiment analysis**

Sentiment analyses are divided into three classification levels: feature level, sentences level, and document level. Document-level sentiment analysis classifies a single document into one polarity. For instance, a review can be classified into positive or negative thoughts. The second level classification is sentence-level sentiment analysis. A single sentence will be classified into one polarity and calculate the total number of positive and negative polarities in the review to consider the polarity of the whole review. Aspect or entity-level sentiment analysis is different compared to these two levels. The first step to finding aspect in the opinion is to determine the aspect of the object in the sentences. For instance, “The quality of the football is quite good” in this example, “quality” is the sentencing aspect. Aspect level sentiment analysis is more challenging because single sentences could appear in more than one aspect, or the element did not determine explicitly.

### **1.5.3 Process of sentimental analysis**

The process of sentimental analysis can be summarized into three steps. The first step is data pre-processing. Some standard techniques are the removal of stop words, which indicates negation. For Chinese deep learning, the word embedding, and tokenization steps are complete in this step. Chinese text makes it harder to do tokenization because Chinese words do not separate by space. Second step is feature selection, feature selection is critical for machine learning methods such as SVM and Naïve Bayes. For the deep learning method, the feature selection process done by the model itself. Third steps are classification, the sentences or document are classified into two polarities.

#### **1.5.4 Introduction to classification method**

There are two big branch classification methods in sentiment analysis: lexicon-based and machine learning-based. The lexicon-based approach uses a lexicon or corpus to find the opinion word and its value from the lexicon. It is the easiest way to use the existing corpus and lexicon resource. Unlike English, Chinese sentiment lexicon and corpus recourse are less than English. For machine learning-based, there are two types of models which are machine learning and deep learning. The main difference between the machine learning method deep learning method is machine is deep learning method can build a more complex non-linear plane in the data space. Deep learning models are more complex and need more time to train. The machine learning method can also be classified as supervised learning and unsupervised learning. Unsupervised learning is a model learning feature from unable data. The deep learning model that gains attention in Sentiment analysis is LSTM, Elmo, and Bert. They outperform other frameworks by attention mechanism ability to learn the decency and deep bidirectional representations of the sentences.

#### **1.6 Report Organization**

This report composed of 7 chapters: Chapter 1 Introduction, Chapter 2 Literature Review, Chapter 3 System Model, Chapter 4 Data per-process and Model output, Chapter 5 Experiment, Chapter 6 System Evaluation and Discussion and Chapter 7 Conclusion. In the first chapter has determine the scope of this project and it objective. In the background information had provided history information until current methodology. At Chapter is discussion of other project work, this project analysis all methodology to propose a new approach. In chapter 3 give general information of specification of methodology and an overview of the model. Chapter 4 provided detail on the input and output of the model. Next, Chapter 5 shows the result of the experiment, the setup of the experiment. Chapter 6 is a discussion on the chapter 5 experiment result and concluding the weakness of the model. Last, Chapter 7 give conclusion to this project.

# Chapter 2

## Literature Review

### 2.1 Sentimental analysis in general

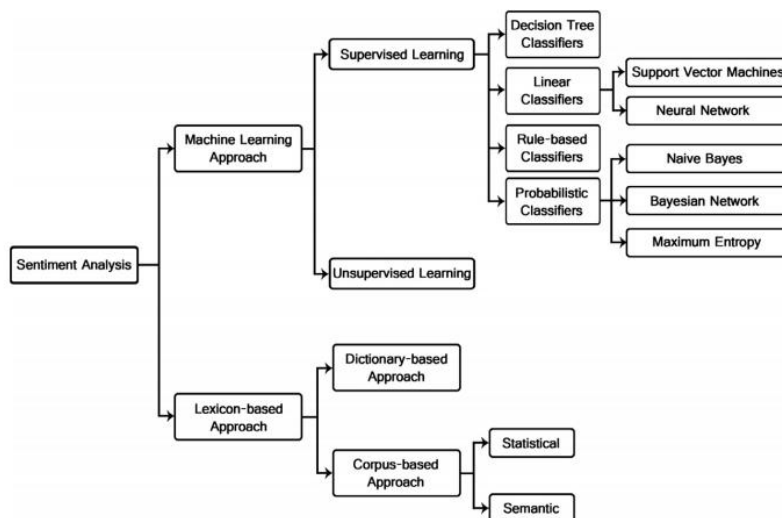
#### 2.1.1 Feature selection on sentimental analysis

Sentimental analysis is a classification problem usually classified into two polarities or three, including neutral to determine an objective opinion. Before classification, some features can extract from the input for the model to work better. Data pre-processing can reduce the model's work and training time to find the precise feature of the model. According to Medhat [7], a few feature selection techniques are:

- Term presence and frequency.
- Parts of speech tagging.
- Opinion words extraction.
- Negation.

Terms presence and frequency count the words and use frequency weight to indicate the relative importance of the feature. POS (Part of Speech) tagging is tagging the sentences and making a model that can differentiate nouns, adverbs, verbs, and adjectives. Opinion words extraction uses lexicon or corpus to find those adjective and sentimental related words, for instance, good and bad. Negations will change the original context, which must be pre-processed in some models.

### 2.1.2 Technique on sentimental analysis



*Figure 2.1.1 Sentimental Analysis Techniques [7]*

The sentimental analysis classifies document level and sentence level text into positive and negative opinions. It is different from feature-based opinion mining because it considers the whole document as a primary information unit. Overall sentiment polarity will be calculated to classify the text into other classes. Document-level and sentence level classification do not differ much. Sentimental analysis can divide into two different techniques: machine learning and lexicon-based approaches. The combined technique is used in some frameworks to obtain higher precision and achieve their objective.

Machine learning has two methodologies which are supervised and unsupervised learning. The difference between the supervised and unsupervised methods is that the supervised approach uses a tagged training document, whereas the unsupervised method does not. Pang and Lee [9] employed the Naive Bayes Classifier, maximum entropy classification, and support vector machines to assess the movie review. Pang and Lee believe this dataset to be challenging because it is cross-domains [9]. Throughout 120 documents, they received an average score of 65.83. Pang and Lee avoid an uneven dataset by limiting each contributor to 20 reviews. Naive Bayes works well on problems with highly dependent characteristics [9]. Maximum entropy is different from naive Bayes in that it ignores feature relationships. According to Pang & Lee [9], SVM has the best accuracy of 82.9 with unigram feature extraction, but he can't

match standard topic-based categorization accuracy. Although balancing the class distribution in training data can help the model perform better, it can harm Naive Bayes outputs [7]. According to a survey conducted by Thakkar and Patel [14], the accuracy of several methodologies for sentiment analysis on Twitter ranges from 63 percent to 80 percent. The feature selection is the essential component that influences accuracy. In short machine learning approach need to select relevant features and train on balance datasets to obtain high accuracy.

An opinion lexicon is used to examine the text in a lexicon-based method. The performance of these systems is dependent on high-quality, wide-coverage lexical resources [8]. A positive opinion word conveys a positive attitude, while a negative opinion word conveys a negative attitude. Dictionary-based opinion words are frequently divided into positive and negative sets using WordNet to locate synonyms and antonyms for each adjective. A corpus-based approach measures co-occurrence patterns and uses a seed list of opinion terms annotated by humans early in finding opinion words in a huge corpus. "You are lovely and stunning," the time "lovely" can be deduced from the word "beautiful," which is already a well-known favorable opinion word. The dictionary-based technique has the drawback of being unable to find the domain and context-specific opinion terms. The problem of finding opinion words with context-specific orientation can be solved using a corpus-based approach [7]. When the level of specialization rises, sentimental analysis falls short. Moreno-Ortiz & Fernández-Cruz [8] apply the "plug-in" lexical resources to deal with a single domain. His idea is to extract candidate terms from specialized corpora and match them against the general-language polarity database to obtain orientation for domain-specific corpus and use some rules to adjust the polarity and handle some cases. For example, if the candidate phrase exists in the general-language lexicon, it is added to the specialized sentiment lexicon. Their method yielded an accuracy score of 84.21. To deal with a different domain, change the specialized corpora to the related domain. A hand-tagged lexicon of adjectives may be required to achieve acceptable accuracy. The performance of lexical analysis degrades as the dictionary grows larger.

### **2.1.3 Deep Learning in Sentimental Analysis**

Machine learning is an iterative process that use data and algorithm to produce a reliable model. Opinion mining in machine learning is recently one of the most studied approaches to sentiment analysis. Classification models such as Support Vector Machine and Naïve Bayes were used to classify text's polarity. An accuracy score of 82.9% was reached by Pang et al. [9] using Support Vector Machine and unigram feature extraction. Still, this framework cannot work well in cross-domain and cross-language sentiment analysis. Deep learning is considered state-of-the-art in computer vision and speech recognition [11].

Deep learning in sentiment analysis did not get a significant or remarkable improvement before the Bert Model. Zatarain Cabada [24] et al. used the combination of Convolutional Neural Network and Long Short-term memory for opinion mining, and the accuracy score is 92.15. One of the significant differences between the Machine learning approach and Deep learning is feature engineering. Machine Learning needs to use statistical techniques to analyze the relationship between data, but Deep learning will automatically get the connection through its network. The sentimental analysis with the machine learning approach cannot get high precision when related to cross-domain or sentimental analysis in a foreign language. In conclusion, the deep learning approach is better than the lexicon-based and supervised learning model, but it takes more time to train. The accuracy of deep learning is directly proportional to the data size to train.

## **2.2 Chinese text sentimental analysis with deep learning approach**

Currently, the deep learning model is one of the most popular models because it auto feature engineering and has high accuracy compared to other methods. Among many deep learning models, the Bert model is the best compared to other NLP tasks. Bert model has the best performance in terms of understanding the context of the sentences compared to LSTM. Bert model also makes the training process parallelizable with whole input sentences into the model. This section will discuss different deep learning frameworks in Chinese sentimental analysis and why they cannot beat the Bert model.

### 2.2.1 Convolutional neural network

A convolution neural network is one type of neural network that performs well in the image-related task. It will take the input in a 2d or 3d matrix and shrink the output size after layer and double up the channel until the size of the output cannot be further decreased and connected to a fully connected layer. Many researchers try CNN for Chinese Sentimental Analysis but cannot outperform sequential deep learning models like LSTM. Liu [23] collected its own dataset by the crawler and used a convolution layer to extract features explicitly and used SVM and RNN to classify the sentimental. This framework wastes the feature extracted from CNN and uses RNN as a classify output from the convolution model, which is not sequential data is not good practice to do deep learning classification model.

Wang et al. [17] proposed a Char-CNN with a Support Vector Machine as a classifier. They believe most of the researchers do not pay attention to the Chinese characters. The framework takes Chinese character with tone as input and conclude that even though Char-CNN cannot understand the semantic structure but still can produce a good result. Technically, this model is not good because many Chinese characters with have same pinyin and tone can make ambiguous semantics, and single characters usually are neutral and do not have any polarity.

Xu Wang et al. [16] believe that Chinese characters and words can bring semantic meaning to the polarity. They use two streams as input which are words and characters and use one-hot encoding to mark them in the vector space and put them into two CNN to extract the feature. The result of two CNN is concatenated together and passed into the SoftMax layer. They experiment on the Chinese sentiment corpus and conclude two streams of improvement do improve the accuracy but are diminutive. Yu Zhang et al. [26] also proposed a similar framework with a two-stream of CNN and taking word embedding feature and sentimental feature vector. They also show a similar conclusion compared to the Xu Wang project but also show diminutive improvement only. Xiao et al. [19] proposed a single stream CNN, but concatenated vector from word vector and character vector draw a similar conclusion. As a result, dual-channel gets 0.3 % improvement compared to single-channel CNN.

Feng Xu [20] uses CNN on the ChnSentiCorp dataset and gets around 98% accuracy. Although he does split the dataset to test set and training set, he did not mention any information about 98 % is test accuracy or training accuracy. The model may have an overfitting issue because the regularization method he applied is just normalization. Basic CNN gets satisfactory performance on image processing tasks, but it cannot be trained too deep because deep networks are prone to overfit, and the different sizes of the kernel on CNN can get more information if the information is distributed every scatter. Zheng Xiao et al. [18] get the idea from GoogLeNet (Inception CNN), which has a different kernel at the convolution layer. He proposed a convolution control block and trained his model on Mio-Chn-Corp (Million Chinese Hotel Review) dataset. He concludes that the model is the best when compared to the Linear Regression model and Deep & Cross network model, and that is not enough.

Recently attention-based model has state-of-the-art performance in the speech recognition task. Li Yang [21] proposed a CNN with an attention-based Bidirectional Gated Recurrent Unit. The framework proposed use sentiment lexicon and deep learning technology to solve some issue of the existing analysis model. After the word has been embedded into the matrix, it will forward to the convolution and pooling layer. After that is a Bi-GRU, attention layer, and fully connected layer. The Novelty is they extract context features from the input matrix with Bi-GRU and use the attention layer to find the exact feature which can determine the polarity of the sentences. In a short conclusion, the CNN model is not suitable for the sentimental analysis model. According to the above framework, they cannot prove that their model also works well in another dataset or other domain. LSTM can get a better result than CNN because the hidden state carries information from previous input, which is an important feature.

### **2.2.2 Long Short-Term Memory**

Long Short-Term memory is an improved framework based on the drawback of RNN (Recurrent Neural Network). RNN is a kind of multilayer feedback neural network because it takes one input at a time and calculates the previous hidden state and takes new timestamp input to calculate the new hidden state until the last input [28]. Based on how RNN worked, it is suitable to process sequential type data because the current hidden state is based on the previous hidden state. It can understand the previous context and put it into a feature in terms of a hidden state. However, RNN has two



critical drawbacks, it has the vanishing, and exploding gradient problem, which comes from the sigmoid activation function, and RNN cannot remember the beginning context when taking an exceedingly long sequential input. These two critical limitations make the researcher explore an improved version of RNN, which is LSTM which has a gated component to decide what to forget, input and output. Although LSTM solves vanishing gradients and can retain information, it also has its drawback in terms of cannot being parallelizable and hard to implement because it is hard to train the model.

According to Zhang [26] conducted an LSTM experiment on an e-commerce platform review, different LSTMs with different model parameters, such as input length, can lead to a different result, and the parameter should be adjusted to achieve the best result for this dataset. He adjusted the parameter and did not test on another dataset, do not convince this was a good framework. According to Day [2] she uses three dictionaries which are How Net, NTUSD, and iGoPaSD, to integrate opinion words and pass them into Word2Vec to generate word embedding from google play consumer review data. She proves that her model Bi-LSTM can beat Naïve Bayes and SVM, which is reasonable because Bi-LSTM can extract features by itself, but NB and SVM cannot.

LSTM model is one direction, and researchers argue that one direction input cannot make the model understand the context of the sentences and words, so they proposed Bi-LSTM (Bidirectional LSTM), which is two LSTM takes input stream from start to end and from end to start. Yang [22] discovered the Elmo feature on word embedding, which is a Bi-LSTM model doing word embedding tasks. He used the Elmo model for word embedding and RNN as a classifier to do sentimental analysis on dataset crawl from Jindong (e-commerce platform) and concluded Elmo does do an excellent job on word embedding. From Chen's [1] analysis of LSTM on the Chinese amazon website review, she makes the conclusion LSTM model has good learning capability compared to SVM, Naïve Bayes, and other non-machine learning approaches. She also concludes Bi-LSTM model can improve accuracy but is diminutive, which is 1 % only.

Kai Zhou [27] uses a CNN to shrink the input size and pass to two Bi-LSTM for his model architecture. He compared his model with other models such as CNN,

LSTM and Bi-LSTM also got a similar result in which the accuracy difference of each model has only 1%. The difference was shown when using word2vec as the word embedding layer, so it can be concluded that most of the feature was extracted in the word embedding layer, which is not from the architecture he proposed. According to Zhou [25], the CBOW (Continuous Bag of Word) model and stacked Bi-LSTM achieve a better performance which is around a 1.6% of improvement. When Stacking LSTM improves, the accuracy also improves the training time because its parameter will be doubled. The trade-off return is not fascinating. Stacking LSTM also does not seem to be an effective way to improve the model architecture.

Recently attention mechanism was proposed in image processing with the intention people will focus on certain areas of the pictures. People also have similar behavior try to understand the context of a sentence. People tend to focus on certain keywords and decide the polarity of the sentences [10]. Su [10] proposed architecture with convolution in this first module and the output forward to Bi-LSTM, which is the second layer, and use attention layer as the third layer and connect to fully connected layer. Many researchers have started to focus on word embedding instead of the classification model. Good word embedding can affect the classifier model because word embedding is getting the context of the word. This framework above, which only focuses on one specific task, cannot be implemented in real life because it only fine tune for the specific task and dataset. People started to research the unsupervised model, which has a better ability to understand the language so it can be applied in all NLP tasks. There are two objectives which are to make it can be unsupervised learning and can be used as a pre-training model to do transfer learning for quick implementation.

### **2.2.3 BERT**

Recently, the pre-training model has become a research trend because it is effective in improving NLP tasks [3]. BERT model is known as Bidirectional Encoder Representation from Transformer. Bert model architecture is inspired Transformer model, which takes the encoder and makes some modifications. According to Devlin [3], BERT is designed as a pre-training model by learning bidirectional representations from the unlabelled text by combining left and right contexts in all layers. To transfer learning, they are two approaches a feature-based such as Elmo, or fine-tuning approach such as OpenAI GPT (Generative Pre-trained Transformer). Devlin argues that OpenAI

GPT and Elmo are unidirectional or use one-directional, which restricts the power of the pre-training model. To achieve a deep Bidirectional encoder. The Bert model will mask some of the tokens and predict the token based on the context. It will use the attention mechanism, which allows the prediction is based on the whole representation. Unlike Bi-LSTM, left to right or right to left would allow the model to be indirect and see “itself” from another direction [3]. Additionally, the BERT model not only does Masked token but also predicts the next sentence by putting two consequence sentences, and the model will guess which one is the before and after. This is the pre-training step, for fine-tuning, BERT will initialize with pre-trained parameters and fine-tune with label data. On the GLUE benchmark, the BERT model is outperformance Elmo and OpenAI GPT. GLUE benchmark is testing the model on natural language understanding. A model which scores well in the GLUE benchmark can easily apply all NLP tasks with satisfactory results.

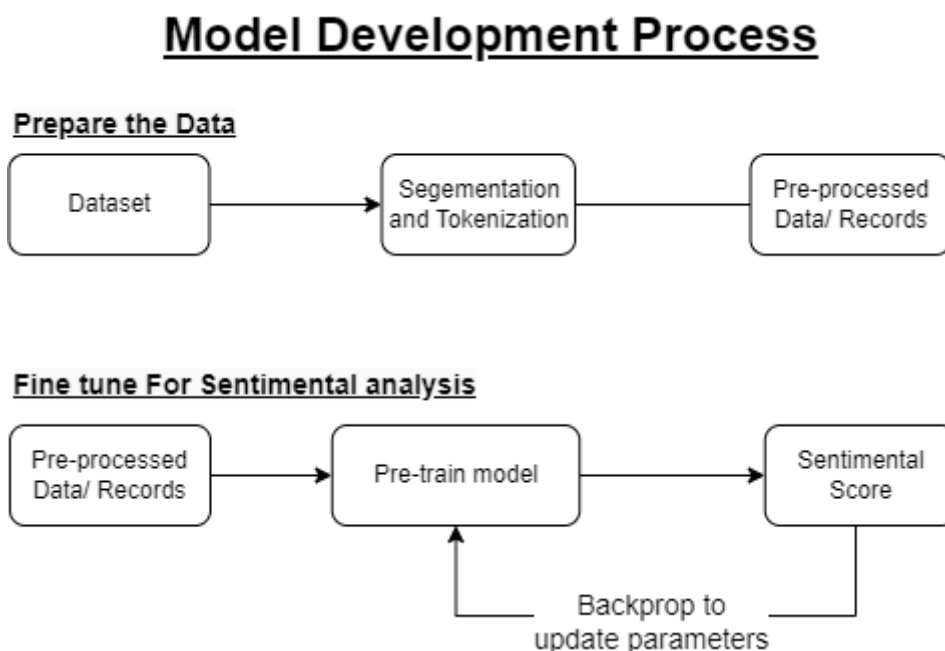
BERT model also has been applied to Chinese Sentimental Analysis. Chinese sentence orientation is different from English. Word embedding is the most important part of the NLP task. The ZEN model from Diao [4] and ERNIE model from Sun [12] make a difference in how the models do word embeddings and masking. ERNIE models use different levels of masking, such as basic-level, entity-level, and phrase-level masking, which is better than the original BERT Model. The ZEN model has a two-level encoder which is character level and n-gram level. N-gram feature extraction by use of an n-gram lexicon. The n-gram embedding will further add to the character level by an n-gram matching matrix. Sun [12] claimed that their ZEN model is better than the ERNIE model because the ERINE model takes more datasets to train compared to the ZEN model. ERINE model is better than Bert by implementing different level masking and changing the Masked language model to the Dialog language model. They prove their model can guess the correct entity, but Bert's model only can get the correct pattern only. Although the Bert model is good enough, without the next sentence prediction task, Bert also can score well, which is an experiment by Roberta (A Robustly Optimized BERT Pretraining Approach) [6]. This project will explore the best approach to do word embedding by either focusing on masking or sequence prediction.

# Chapter 3

## System Model

### 3.1 Overview of the Model and development process

The project goal is to achieve high accuracy on Chinese sentimental analysis. Based on the research, the best solution is using a pre-train language model which has understand the context of language and words and fine tune it the language model to do completed specific task. Ernie is one of the best language models which also original trained in Chinese language task. In the project we will use Ernie pretrained model to fine tune it for sentimental analysis task.



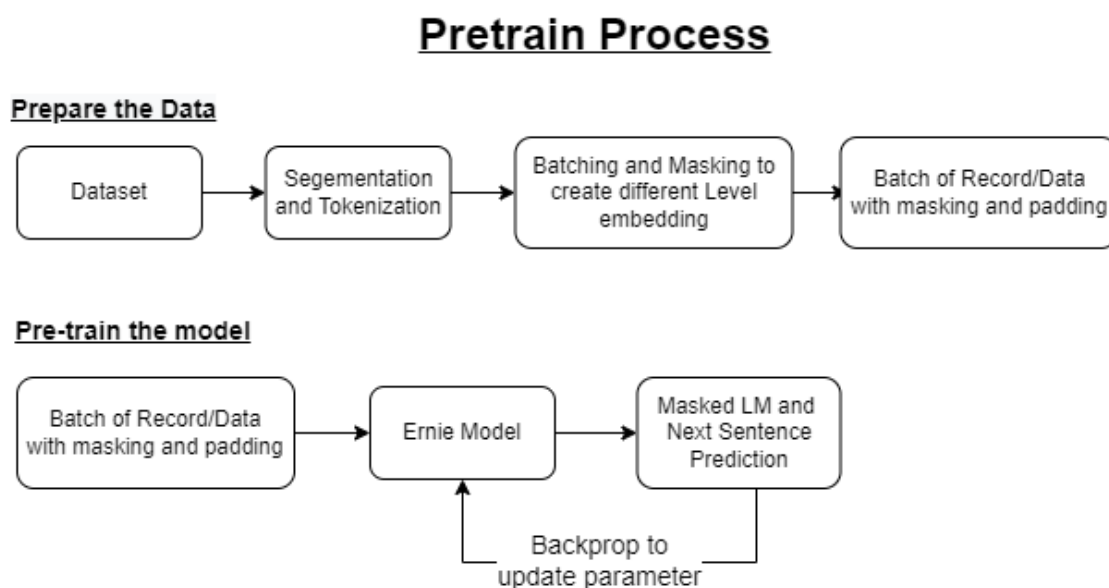
*Figure 3.1 Model Development Process*

This project model development process can be divided into two process which is data preparation and fine tune process. The dataset records are unprocessed sentences will be doing data cleaning for example punctuation removal and then the cleaned data will be doing word segmentation by WordPiece algorithm. After segmentation, the data will be converted to token by refer a vocab text file which from the Ernie pre-trained model. Next the token will combine with other information tokens such as position token, sentences type token. After tokenization process data will be batch into groups.

In a batch of records, the records will be padded to reach longest length so the records will have the same shape and length. For the fine tune process, single batch of records will input into the Ernie pre-train model to output single batch of sentiment score. Those sentiment score will be used calculate loss to backpropagation and update the model parameters.

### 3.2 Pre-train Model explain

Since this project use a pre-train language model. This chapter will supply explain on the pretrain process of the Ernie language model. The pre-train process is targeted to understand the language context. The train task is design by Sun.al [12] which is dialog prediction and masked prediction.



**Figure 3.2 Pretrain process**

The pre train model development process like this project model development process but with more procedure and change the target to be predict. After segmentation and tokenization, the records will be batch into group first and padding to match the longest sentences. After that, the records will be applied different level of masking by changing the token to mask token “[MASK]”. Parts of processed records will pair with random sentences to make a compound sentence to make negative sample. The processed dataset will input into the model to predict the sample is original dialog or fake generated dialog, the model also will predict the masked token its original word. The output of model will be backpropagation to calculate parameter to be updates.

### 3.3 Model framework and framework version

The model is built with PaddlePaddle framework. PaddlePaddle one of the deep learning frameworks like Pytorch. This model is built on PaddlePaddle 1.7.2 version. First version of Ernie is built on PaddlePaddle version 1.5, and paddle 1.5 is only supply static graph. PaddlePaddle 1.7 has added Dygraph mode but is disable by default. Dygraph make programmer easier to test the model and debug process also quicker. The project model will be implemented on Baidu AI Studio. Baidu AI Studio an online Jupyter notebook will supply graphics processing unit to make ease of user learn and run model online. Baidu AI Studio is one of best online Jupyter notebook to do machine learning. It supplies popular dataset and set up of the environment is easier compared to run the model on local machine.

### 3.4 Dataset

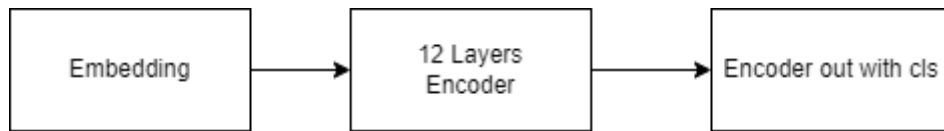
The dataset that used for fine-tuning process is ChenSentiCorp. It is a hotel review data set collected by Tan Songbo. There are 10000 review is the dataset. In between 10000 records, 5000 are positive ,2000 are negative and remaining are unlabelled samples. This project gets the dataset resource from is from Paddlenlp module. In the module the dataset has been combined, change the label and remove duplicate samples. The dataset combines of two columns and one label.

	label	review
5612	0	房间小得无法想象,建议个子大的不要选择,一般的睡觉脚也伸不直,房间不超过10平方,彩电是14...
7321	0	我们一家人带孩子去过“五一”,在携程网上挑了半天才选中的酒店,但看来还是错了。1.酒店除了...
3870	1	周六到西山去采橘子,路过这家酒店的时候就觉得应该不错的,采好橘子回来天也晚了,就临时决定住在...
4057	1	交通很便利,到渔人码头和港澳码头都在步行的范围之内,CHECKIN和CHECKOUT的速度都...
1452	1	很不错的一个酒店,床很大,很舒服,酒店员工的服务态度很亲切。
4805	1	酒店环境和服务都还不错,地理位置也不错,尤其是酒店北面的川北凉粉确实好吃,不过就是隔音效果不...
6868	0	旧楼改建的酒店,期望不要太高。酒店经理的态度很好,会帮助解决问题。有一位前台小姐的态度实在是...
1345	1	经常去海口出差,但从没住过该酒店,看外表感觉一般吧其实酒店里面还真不错,房间是新装修的(我住...

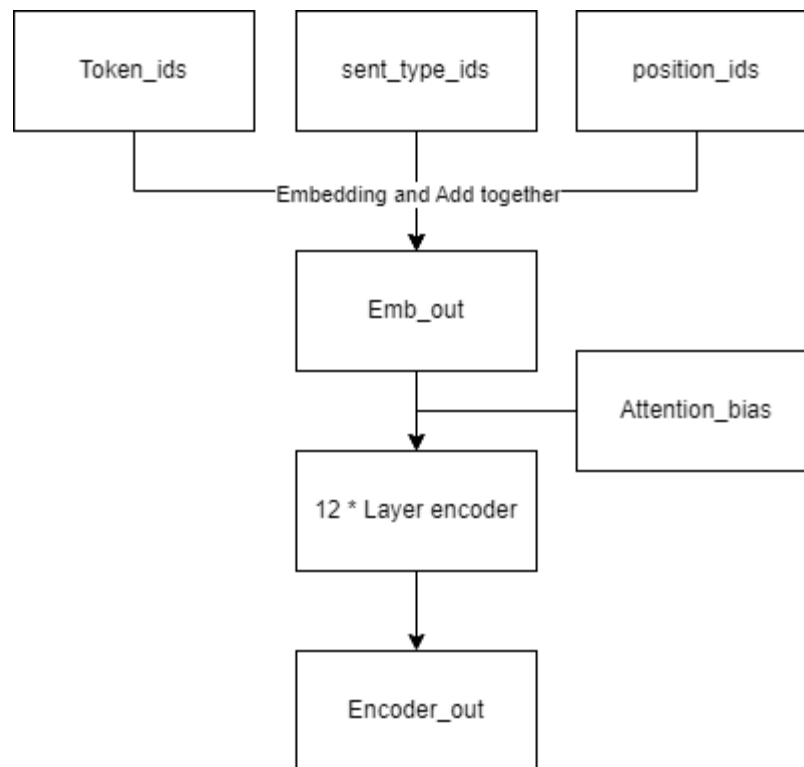
*Figure 3.4 Records of the ChnSentiCorp*

The first column is query id which will be not used to be feature of the dataset. Another column is the review. In a single review combine of multiple sentences and with the punctuation.

### 3.5 Model Design



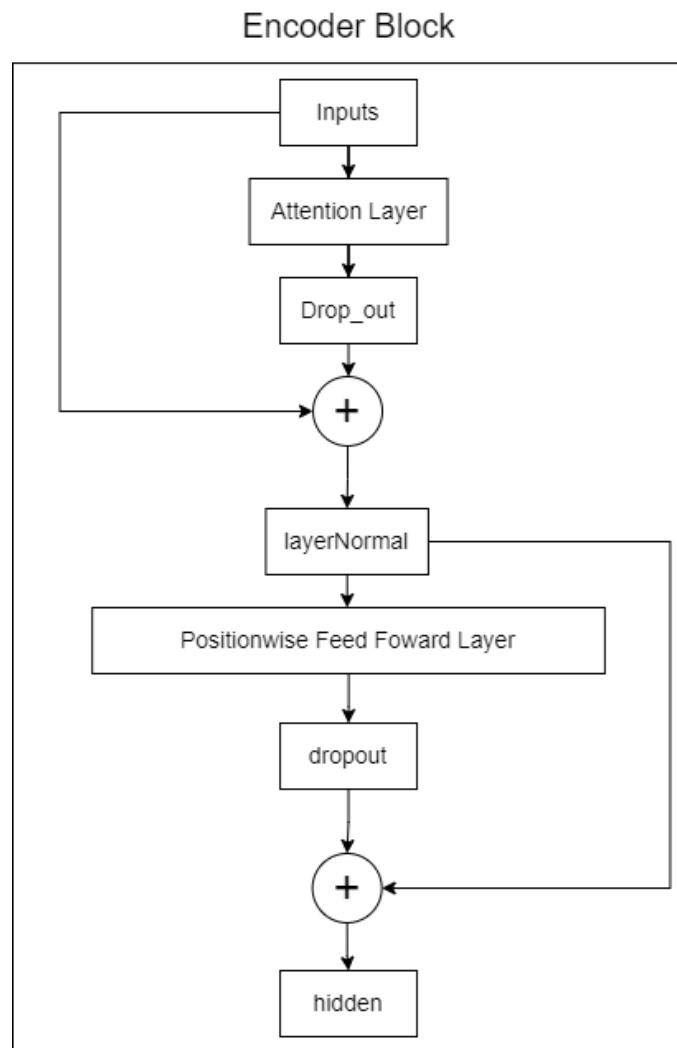
*Figure 3.5.1 Ernie Model in general*



*Figure 3.5.2 Ernie Model input and output*

The Model is made up from stacking 12 layers of encoder. Before input into encoder the tokens need to send to embedding layer. The embedding layer will convert token into fixed length vector based on hidden size. Attention bias is the second input need for the encoder. For sentimental analysis, Attention Bias is not important because attention bias decide which token will be masked. It is use for other training process, for example pre-train. The output of the model is a list of tokens and a classification score which is from first token, the “[CLS]” token.

### 3.5.1 Encoder

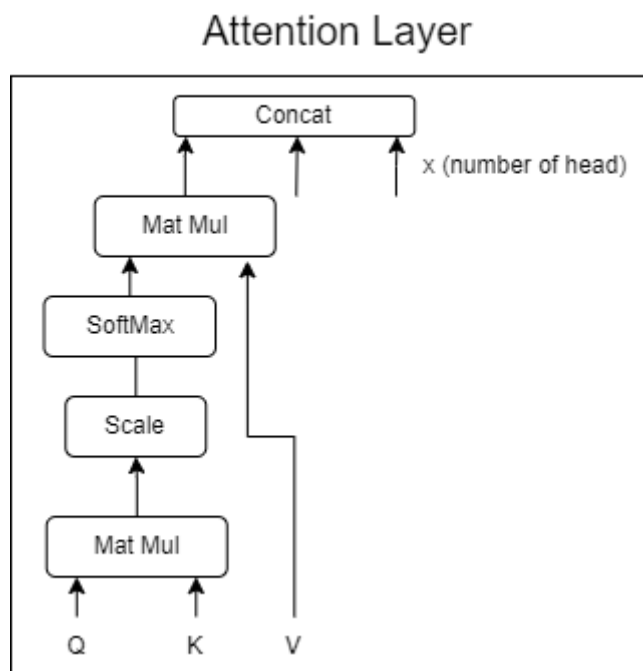


**Figure 3.5.3 An Encoder Block**

A single encoder is made up from one Attention Layer and Position Wise Feed Forward Layer. Attention Layer give the model attention mechanism and Position Feed Forward Layer is a fully connected layer. Between these two modules the layer connected with the dropout layer, normalization layer and residual connection. Dropout layers avoid the model overfitting by randomly dropout inputs. The residual connection and the normalization process which use to avoid parameter weight gradient vanishing and exploding problem. This encoder is from Ernie model and Bert Model which inspired by transformer model.



### 3.5.2 Attention Layer

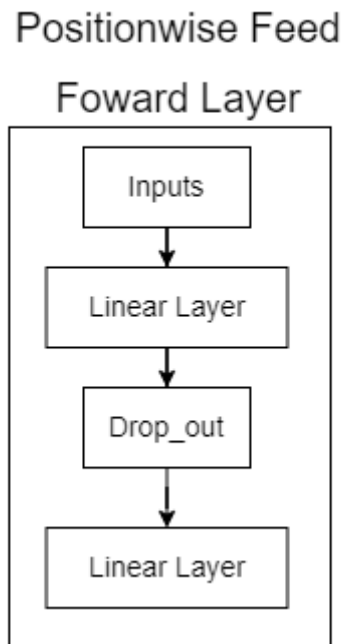


*Figure 3.5.4 An Attention Layer*

$$Attention(Q, V, K) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Attention Layer can make model has attention mechanism, the attention mechanism work like human. When a person needs to understand the meaning of a sentence, he will look for specific words in the sentence. The Attention Layer give the model attention mechanism to know connection between the entity, words and understand the meaning of the words. These Attention Functions take three input which is Query(Q), Key-Value(K), and Output Value(V). Q, K, and V are the same input as the embeddings token in this task. The matrix of Q and K will produce an Attention filter, a matrix with a score on how related these words are to other words. Then, the filter will be scaled down and passed into SoftMax to get the score has the value range of 0 to 1. Attention Filter will be multiple with the V (the original value) to get the filtered value. Multi-Head Attention is used to make the encoder learn multiple linguistic features, and each Head learns one type of linguistic pattern.

### 3.5.3 Position Wise Feed Forward Layer



***Figure 3.5.5 A Point Wise Feed Forward Layer***

This Layer is used to supply the model non-linearity. The Point Wise Feed Forward layer compose of two linear layers, and one drop out layer. The linear layer is using ReLU as the activation function. ReLU activation can avoid vanishing gradient problem and faster computation compared to other activation. It provides non-linearity and good for Ernie which stack 12 layers of encoder and each layer of encoder has two linear layers.

## Chapter 4

### Data per-process and Model output

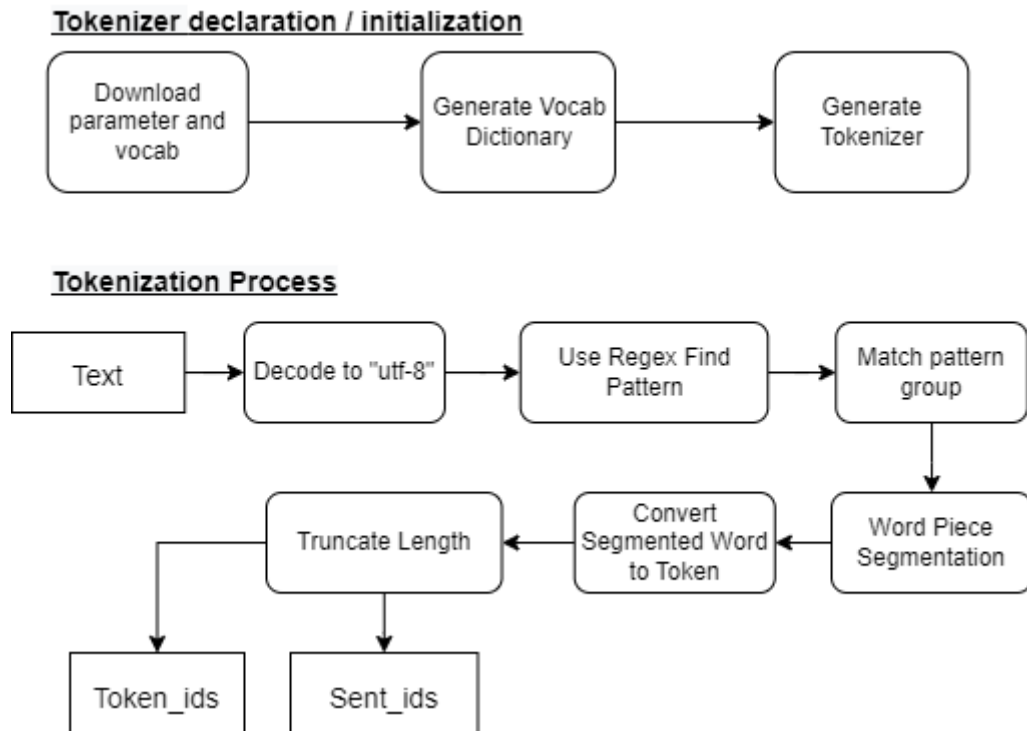
#### 4.1 Tokenization Process

The tokenization process is one of the important steps because it prepares the data for the Ernie model. Ernie model is a language model, the text needs to be cleaned invalid characters, segmentation, truncate length so it does not exceed maximum length. The maximum length Ernie model can take is 512. If the document words can exceed the maximum length. The tokenizer will remove the exceed part.

1	[PAD]	0
2	[CLS]	1
3	[SEP]	2
4	[MASK]	3
5	,	4
6	的	5
7	、	6
8	一	7
9	人	8
10	有	9
11	是	10
12	在	11

*Figure 4.1.1 Vocab.txt*

For Ernie tokenizer has its parameter, for instance vocab.txt which list down words and its token indexes. Other parameters are encoding format, Boolean flags on conversion of uppercase English word and token string declaration. In the vocab.txt number 1 to 4 words is the special token which has its context to the Ernie model. The “[PAD]” token is for padding; it use to make sure in a batch of records has same length of the text data. The “[CLS]” is added at the start of the text to indicated where the text start. “[SEP]” character is the to indicate start of next sentence. “[MASK]” is used to mask the token for Ernie pre-train process. For fine tune process, the mask token will not be applied.



*Figure 4.1.2 Tokenization Process*

To fully use the knowledge of Ernie pre-train model, the tokenizer must have same parameter and vocab file which use to pre-train the model. The configuration and parameter of the tokenizer is for Ernie 1.0, Ernie 2.0 and Ernie-Tiny is different. The parameter of different Ernie Tokenizer and Ernie model configuration can be access using these URL.

```

URL = 'https://ernie-github.cdn.bcebos.com/'
resource_map = {
    'ernie-1.0': URL + 'model-ernie1.0.1.tar.gz',
    'ernie-2.0-en': URL + 'model-ernie2.0-en.1.tar.gz',
    'ernie-2.0-large-en': URL + 'model-ernie2.0-large-en.1.tar.gz',
    'ernie-tiny': URL + 'model-ernie_tiny.1.tar.gz',
}
  
```

*Figure 4.1.3 Resource Map*

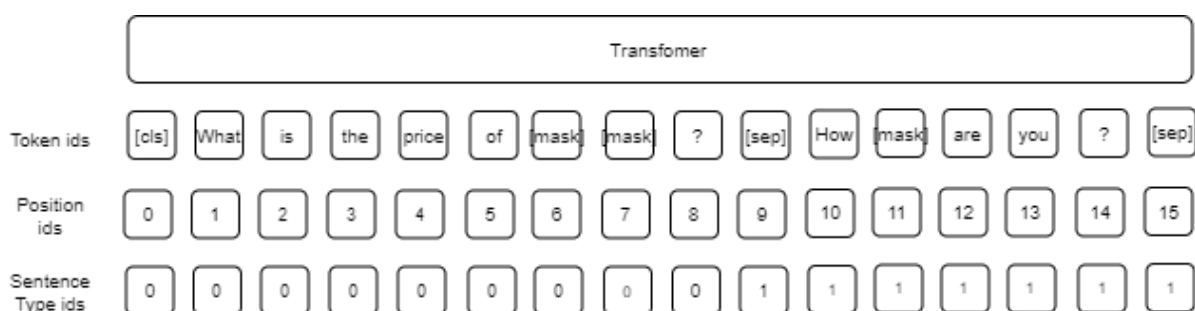
The project will use the resource map above to get the correct tokenizer for Ernie model. After download the parameters and vocab file from the URL, the constructor will read vocab text file and generate a vocab dictionary. Tokenizer initialization is complete once the vocab dictionary and the parameter has been set.

Tokenization process first step is decoding the string to “utf-8”. Since Unicode problem exist the Chinese character encode in “GBK”, it will occur problem if we do not decode the string which has been encode in “GBK”. After the string has been decode, tokenizer will use regular expression to find matching groups. After that, matching groups will be input into WordPiece segmentation algorithm. Texts had transformed to tokens with the WordPiece algorithm. Next, vocab dictionary is used to convert tokens to its indexes. For instance, “[PAD]” convert to 0 and “的” convert to 5. Last, the length of token is truncate the length to 512 and add the special token at start, between the sentences to produce Token indexes and Sentences Type indexes.

```
(\[PAD\])|(\[UNK\])|(\[MASK\])|(\[SEP\])|(\[CLS\])|([a-zA-Z0-9]+\|S)
```

*Figure 4.1.4 Regex pattern for Tokenizer*

## 4.2 Model Input



*Figure 4.2.1 Model input in general view*

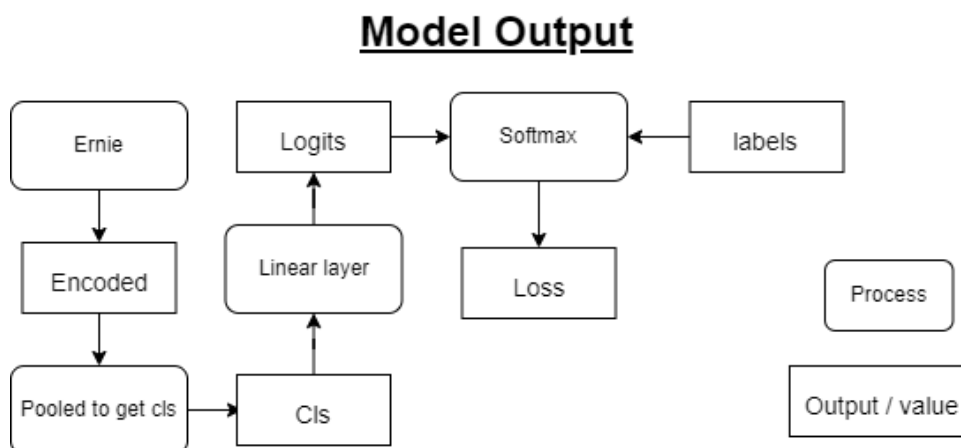
The input of the Ernie model is made up of three types of tokens which are Token ids, Position ids and Sentence Type ids. The Token indexes carry the word context. It will be added with the special token such as “[cls]”, “[sep]”, “[pad]” to indicate the start of first sentence and next sentence. The position ids are the position token which range from 0 to the longest length of tokens in the batch. The sentence type ids are to determine which part is from first sentence and which part is next sentence.

Token_ids	1, 208, 42, ... 42, 2 (From Vocab.txt)
sent_type_ids	0, 0, 0, ... 1, 1 (0 is Sent A, 1 is Sent B)
position_ids	0, 1, 2, ... 61, 62 (label the position)
input mask	1,...,0 (0 is no attn, avoid attn on pad )
attn bias	[batch size, seq len, seq len] (3d version of input mask)
label_id	0 or 1 (depend on task)

**Figure 4.2.2 Model input in actual view**

Except for the three types of tokens, the model takes input mask or attention bias as optional input. The idea of input mask and attention bias is avoided Attention layer out attention on the padding token. Regarding the type of token ids, sentence type ids and position ids are int type tensors. Input mask and Attention are float 32 type tensors.

### 4.3 Model output



**Figure 4.3.1 Model input in actual view**

The Ernie model output encoded vector, the vector will be pooled to get the first value in every vector. That is the classification score. The classification score will input into one linear layer to produce logits. The logits will be process with the label to generate loss. The loss is used to do back propagation and update parameter in the model.

# Chapter 5

## Experiment

### 5.1 Environment Setup

This project use Ernie per-train model to do sentiment analysis fine tuning. The Ernie modeling and tokenization file is from the PaddlePaddle Ernie GitHub repository. The repository has removed, and the latest Ernie repository is Ernie 2.0 which is not the scope of this project. The whole experiment will be conduct online on the Baidu AI Studio. The file of Ernie model and tokenization will be upload into the AI Studio Project. Next step is installing the library which in the requirement.txt. The library needed to run the model and tokenization process are:

- Natural Language Toolkit version 3.4
- NumPy version 1.14.5
- PyZMQ version 18.0.2
- Scikit-learn version 0.20.3
- SciPy version 1.2.1
- Six version 1.11.0
- Sentencepiece version 0.1.8
- PaddlePaddle-Gpu version 1.7.1

AI Studio allow Jupyter notebook to run Linux commands. The ChnSentiCorp dataset can be download by using Linux command “wget” from the URL “<https://ernie-github.cdn.bcebos.com/data-chnsenticorp.tar.gz>”, then unzip using the “tar” command.

### 5.2 Train Parameters

The parameters of the training process included the batch size, epoch, max sequence length and learning rate. The model will be fine tune with batch size of 32 sample and run through 10 epochs. The batch size could be increase to faster the training process, but it is limit by the RAM size on GPU. The max sequence length cannot exceed 512 and, in this project, set the max sequence length to 300. The learning rate

recommended by the Ernie original creator is  $1e-5$ ,  $2e-5$ , and  $5e-5$ , so we choose  $5e-5$  because bigger learning rate will step fastest to the optimal gradient.

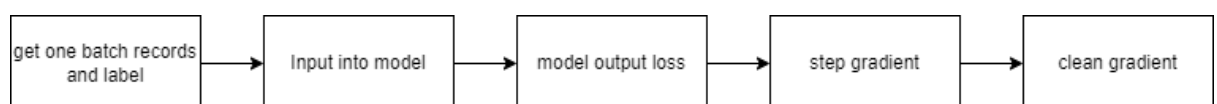
### 5.3 Fine-Tune Process



*Figure 5.3.1 Fine Tune Process*

The Ernie model and tokenizer must be declared first and the pretrained parameter is set to true. The initialize process will download the configuration into cache folder and import the trained parameter to the model. After the tokenizer has been prepared, the dataset is ready to be process. The dataset has been sperate to test set and train set. The record and its label are read by the function one by one. Tokenizer will encode the text to tokens and truncate it to maximum sequence length. After that, all record will pad to maximum sequence length and zip into one variable.

The Fine tune process will loop through 10 epochs. In the interval of training the test process will be conduct after 100 batches samples have been use for training. In the interval of model training and model testing, the information of accuracy metrics mean values and loss mean values was recorded based on epoch.



*Figure 5.3.2 Fine Tune Process2*

The training process start with get one batch of data, input into the Ernie model to get the loss. After that use loss to calculate parameter to be update. Before a new loop start the gradient will be clean. Last, the model parameter that has been train will save into a file. With the file we can load the parameter to do inference without train a new model every time.



## 5.4 Infer with one Sample

```
sample_text="房间小得无法想象,建议个子大的不要选择,一般的睡觉脚也伸不直."
sample_label=0
```

*Figure 5.4.1 Sample Record*

This sample of record is a negative record. The translation of the sample text is “The room is unimaginably small, it is recommended that large people do not choose, the legs cannot be straight when sleeping.” The record need to change it shape and convert to tensor to input into the model.

```
After Tokenization
Token_ids: [  1 458 143  96 116 154  72 313 528  30  81 454  27  85
           19  5  16  41 352 790  30  7 689  5 1695 739 1209 105
          1550 16 339  42  2]

Sent_Type_ids: [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 0 0 0 0 0 0]
Token_ids shape (33,)
label: 0

After Reshape
(1, 33)
(1,)
```

*Figure 5.4.2 Tokenization output and reshape dimension*

The tokenization process will output two types of tokens which is token indexes and sentence type indexes. The original shape of the token is 33 and it need to add the batch size which is one. After the reshape the tensor it dimension become (1,33).

```
name tmp_531, dtype: VarType.FP32 shape: [1, 2]      lod: {}  
    dim: 1, 2  
    layout: NCHW  
    dtype: float  
    data: [9.31006 0.829986]
```

*Figure 5.4.3 Output of the model*

The model output the loss value and logits. Since we one input one sample and the number of classes to classify is 2 which are positive and negative, so the logit has two values only. The logit value determine the possibility of the class. The first value is 9.31 it mean Ernie model classify this sample of record into negative class.

## **5.5 Implementation issues and Challenges**

The training process need GPU to improve the train speed. The model train on 10 epochs will take around 35 mins. To improve the speed to training it need to increase the batch size but the difficult part it the GPU memory can support the batch size. The batch size can be reduced by apply better padding technique. In this project we just pad all record to same size. Better approach can be implemented is pad the record based on the longest length in the batch.

## Chapter 6

# System Evaluation and Discussion

### 6.1 Testing Setup

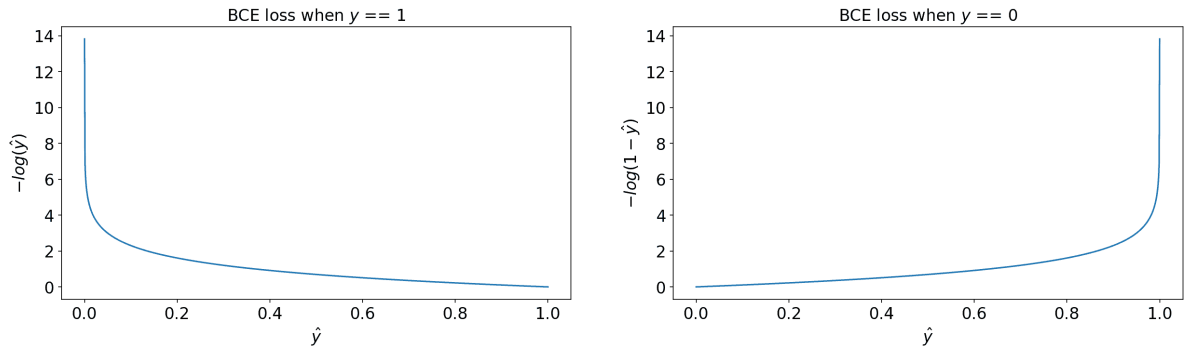
The ChnSentiCorp dataset from the URL had split the dataset into train set and test set. After 100 batches of sample is used to train the model. The test process will be carried out to check the model accuracy. The last model accuracies produce in the last epoch will be calculate it mean and decide to be the accuracy of the model. Other information such as, loss value and F1-scorce will be collected for model evaluation.

Before test process the model need to stop gradient update and set model to evaluation mode. We stop the gradient update so the model will not use the test data to update it parameter. If this step has not be do before test process, the accuracy of the model cannot be used to compare with other projects, because the test set will treat as the train set. The model compose of many normalizations and dropout layers. Evaluation mode will change the behaviors of these two types of layers so precise result can be obtain.

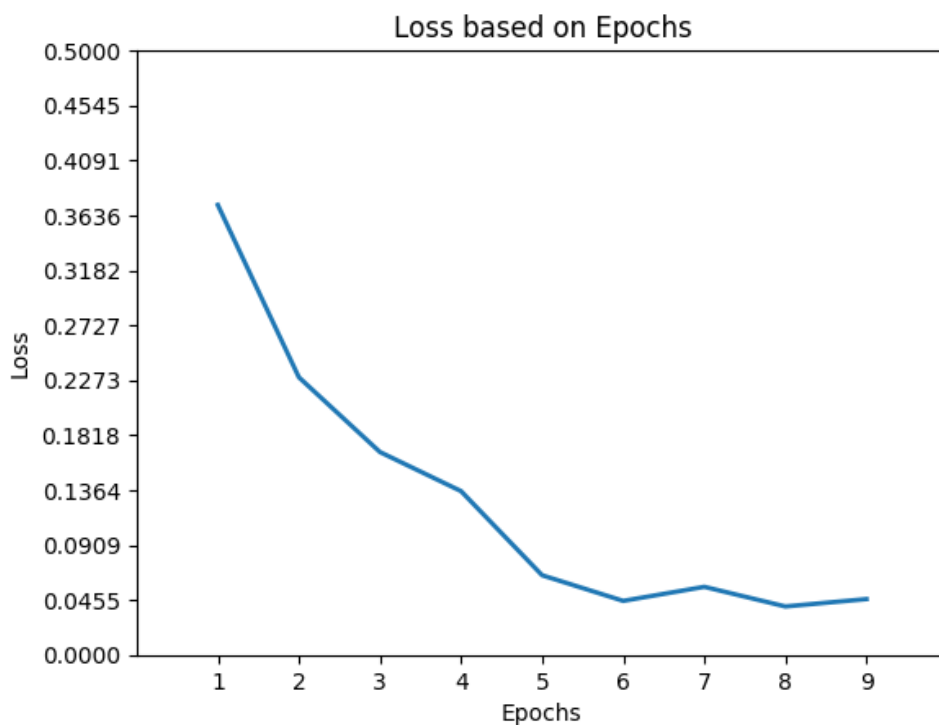
### 6.2 Model Accuracy Metrics

This is classification task, so Binary Cross Entropy loss function is used to calculate the loss value. The Binary Cross Entropy can be used when do binary classification. When the True label  $y$  is 1. The graph on the left-hand side is show loss value depend on the predicted label value.

$$Loss_{BCE} = -\frac{1}{n} \sum_{i=1}^n (Y_i \cdot \log \hat{Y}_i + (1 - Y_i) \cdot \log(1 - \hat{Y}_i))$$

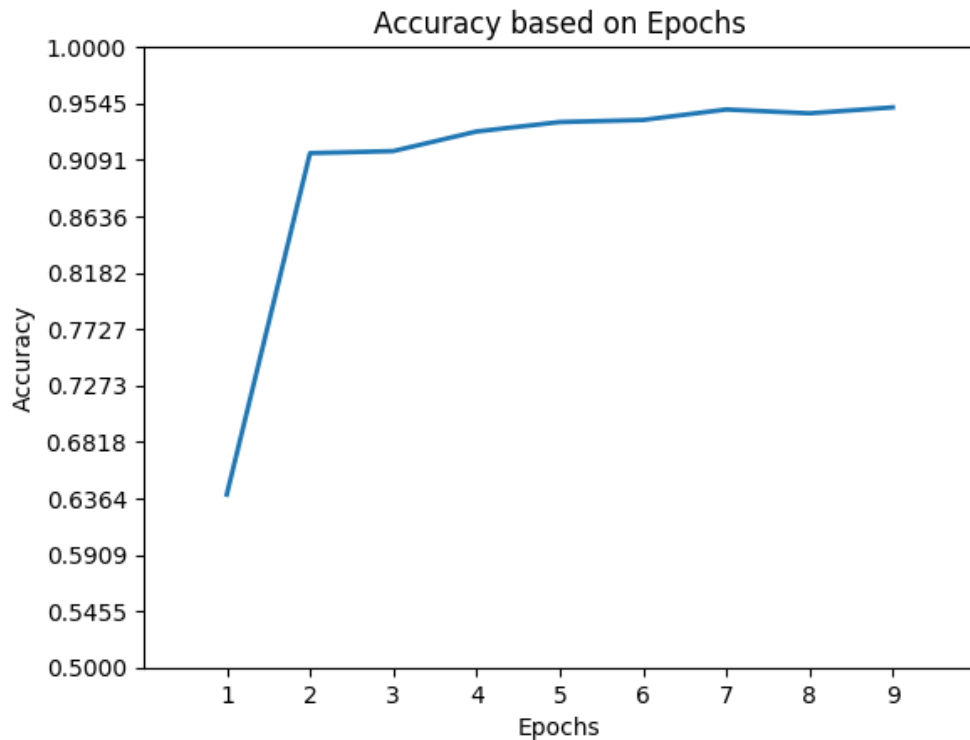


**Figure 6.2.1 Binary Cross Entropy graphs**



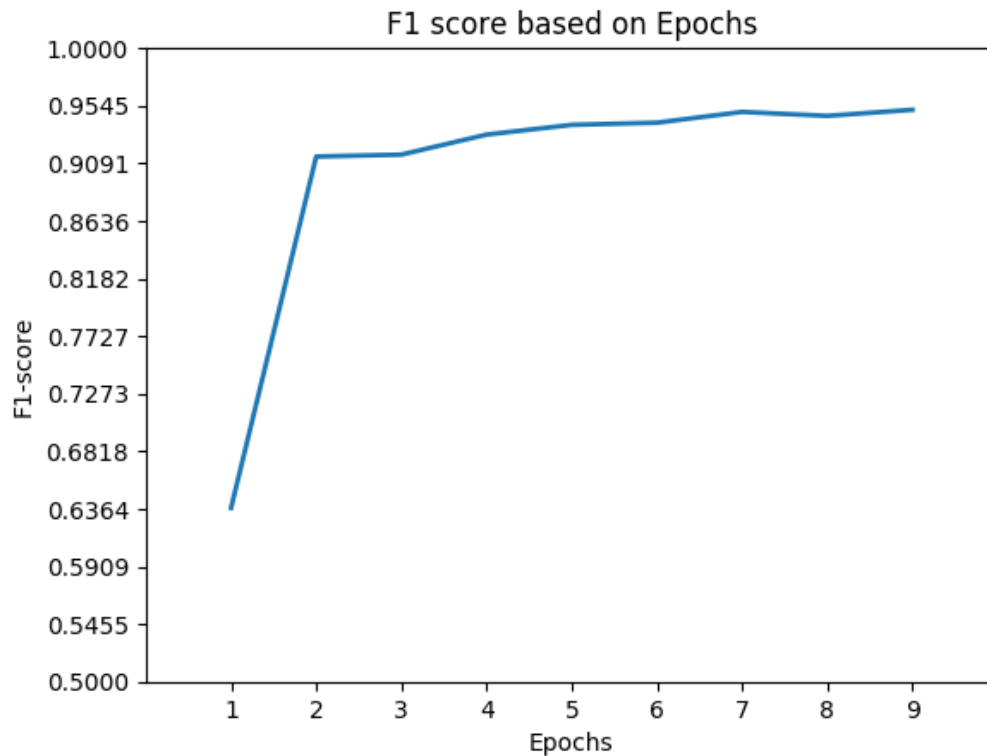
**Figure 6.2.2 Loss based on Epochs line graph**

The loss value of the model is start from 0.4 because the per-train model knowledge help the model can use the knowledge to do classification task. On the epoch 6 the model is completed it training process because on epoch 7,8 and 9 the loss value fluctuate around 0.04. When the model loss value cannot be further reduced, it mean it has completed the training process. The loss value graph could be smoother is 1e-5 learning rate use to optimize the model, but its model gradient will converge at larger epoch number.



***Figure 6.2.3 Accuracy based on Epochs line graph***

The accuracy value start at the 0.6 and grows vigorously at second epoch to 0.9. After that the accuracy value grow slowly when epoch number increase. At the last epoch, the accuracy value reach 0.9513. The accuracy is converged at last epoch and could not be increase and the model training is stop at this epoch. Accuracy could not the best performance matrix because the dataset is unbalanced on the number of positive and negative label. Positive label sample is more than negative label. F1-score can be a better performance matrix on the unbalanced dataset.



**Figure 6.2.4 F1-score based on Epochs line graph**

The F1-score line graph like the accuracy graph. The value of F1-score also stop at 0.9513 is same to the accuracy value. It show that the unbalanced do not affect the training of the model. F1-score calculate the True Positive and True Negative Value unlike the accuracy only calculate the True Positive value. In the nutshell, the model can predict the sample to negative and positive correctly.

The Zen has gotten the F1 score of 95.66 which is higher 0.53 percent different compared to this project model. Ernie 2.0 model get 95.8 on test set accuracy is 0.67 percent which is higher. In conclusion the model accuracy and F1-score different is not more than 1.0 percentage which that show the different of model is not big. To have better comparison of the Chinese Sentiment Analysis task, a complex and newer labeled dataset should be produced for the purpose.

### **6.3 Objectives Evaluation and Concluding Remark**

The objective target to propose a higher accuracy Chinese sentiment analysis. The performance of the model show this project has meet its objectives. The Ernie model which pre-train on Chinese news dataset to learn the Chinese language context and apply different level of masking strategy has shown its improvement compared to Bert model. For application purpose the dataset should be collected from its domain and fine tune the model will get similar accuracy result.

### **6.4 Future work and challenges**

The segmentation algorithm on fine-tune process can be improve by apply better segmentation tool. This project apply WordPiece Segmentation which segment the word to character only. The Ernie model do not release the segmentation tools and the creator has made the statement the Baidu Lexicon Analysis of Chinese can do segmentation result to their segmentation tools. After the segmentation process had been change, the accuracy of the model can be increase.

Ernie 2.0 is having better accuracy because the in the pre-train process it add lot of tasks. Instead of dialog language model and mask token prediction, it made the model train to completed named entity recognition, text similarity, question answering and sentiment analysis task at the same time. It contribute the model learn Chinese language from different approach. Use the Ernie 2.0 model to do sentiment analysis also improve the model performance.

## Chapter 7

# Conclusion and Recommendation

### 7.1 Conclusion

Sentiment Analysis takes one document as input, and its output is a sentiment score. The deep learning approach which can take sequential input is RNN. RNN model has been tried but found that there are exploding gradient and vanishing gradient problems, and LSTM was proposed to solve the issue by introducing a different type of gates. LSTM has the problem that the training process is slow and cannot be parallelized to take whole sentences as input simultaneously. Next transformer encoder with a multi-head attention mechanism starts to get attention to solve the problem of LSTM by using an attention mechanism to focus on more meaningful words and learn the different linguistic patterns. The Bert model gains conception from the transformer and makes the model can learn by Masked LM and NSP, which outperform many tasks in NLP. Ernie models show a better version of Bert by adding a new task, the Dialog language model, and different masking levels make the model learn more. This project absorbs the experience of many approaches to sentiment analysis to propose a better framework so that the objectives can be met.

Ernie pre-train model has shown its power of robust. With the power of pre-train model, the fine tune process to perform sentimental analysis can be done quickly and get high accuracy. Instead of opinion mining, the Ernie pre-train model can be used to do Chinese Language processing task if a good dataset can be obtained.

### 7.2 Recommendation

To improve the model the segmentation algorithm, need to be changed and other Chinese sentimental analysis dataset can be applied to further train the model to improve its performance. Except from ChnSentiCorp Dataset, there was no other dataset has been released for the performance evaluation benchmark. It is encouraged to build for domain related dataset so the application of the model can fit its requirements.



## REFERENCES

- [1] Chen, H., Li, S., Wu, P., Yi, N., Li, S., & Huang, X. (2018). Fine-grained Sentiment Analysis of Chinese Reviews Using LSTM Network. *Journal of Engineering Science & Technology Review*, 11(1).
- [2] Day, M. Y., & Lin, Y. D. (2017, August). Deep learning for sentiment analysis on google play consumer review. In *2017 IEEE international conference on information reuse and integration (IRI)* (pp. 382-388). IEEE.
- [3] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
- [4] Diao, S., Bai, J., Song, Y., Zhang, T., & Wang, Y. (2020). Zen: Pre-training Chinese text encoder enhanced by n-gram representations. *Findings of the Association for Computational Linguistics: EMNLP 2020*.  
<https://doi.org/10.18653/v1/2020.findings-emnlp.425>
- [5] Liu, B. (2012). *Sentence subjectivity and sentiment classification*. *Sentiment Analysis*, 70–89. <https://doi.org/10.1017/cbo9781139084789.005>
- [6] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692.
- [7] Medhat, W., Hassan, A., & Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal*, 5(4), 1093–1113. <https://doi.org/10.1016/j.asej.2014.04.011>
- [8] Moreno-Ortiz, A. & Fernández-Cruz, J., 2015. Identifying Polarity in Financial Texts for Sentiment Analysis: A Corpus-based Approach. *Procedia - Social and Behavioral Sciences*, 198, pp.330–338.
- [9] Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends® in information retrieval*, 2(1–2), 1-135.

## References

- [10] Su, Y. J., Hu, W. C., Jiang, J. H., & Su, R. Y. (2020). A novel LMAEB-CNN model for Chinese microblog sentiment analysis. *The Journal of Supercomputing*, 76(11), 9127-9141.
- [11] Sun, S., Luo, C. & Chen, J., 2017. A review of natural language processing techniques for opinion mining systems. *Information Fusion*, 36, pp.10–25.
- [12] Sun, Y., Wang, S., Li, Y., Feng, S., Chen, X., Zhang, H., ... & Wu, H. (2019). Ernie: Enhanced representation through knowledge integration. arXiv preprint arXiv:1904.09223.
- [13] Songbo Tan, June 23, 2020, "ChnSentiCorp", IEEE Dataport, doi: <https://dx.doi.org/10.21227/yfwt-wr77>.
- [14] Thakkar, H., & Patel, D. (2015). Approaches for sentiment analysis on twitter: A state-of-art study. *arXiv preprint arXiv:1512.01043*.
- [15] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.
- [16] Wang, X., Li, J., Yang, X., Wang, Y., & Sang, Y. (2017). Chinese text sentiment analysis using bilinear character-word convolutional neural networks. In *Proceedings of International Conference on Computer Science and Application Engineering* (pp. 36-43).
- [17] Wang, X., Sheng, Y., Deng, H., & Zhao, Z. (2019). CharCNN-SVM for Chinese text datasets sentiment classification with data augmentation. *International Journal of Innovative Computing, Information and Control*, 15(1), 227-246.
- [18] Xiao (Zheng Xiao), Z., Li, X., Wang, L., Yang, Q., Du, J., & Sangaiah, A. K. (2018). Using convolution control block for Chinese sentiment analysis. *Journal of Parallel and Distributed Computing*, 116, 18–26.  
<https://doi.org/10.1016/j.jpdc.2017.10.018>

## References

- [19] Xiao, K., Zhang, Z., & Wu, J. (2016). Chinese text sentiment analysis based on improved convolutional Neural Networks. *2016 7th IEEE International Conference on Software Engineering and Service Science (ICSESS)*.  
<https://doi.org/10.1109/icsess.2016.7883216>
- [20] Xu, F., Zhang, X., & Xin and Alan Yang, Z. (2019). Investigation on the Chinese text sentiment analysis based on convolutional neural networks in Deep Learning. *Computers, Materials & Continua*, 58(3), 697–709.  
<https://doi.org/10.32604/cmc.2019.05375>
- [21] Yang, L., Li, Y., Wang, J., & Sherratt, R. S. (2020). Sentiment analysis for e-commerce product reviews in Chinese based on sentiment lexicon and deep learning. *IEEE Access*, 8, 23522–23530. <https://doi.org/10.1109/access.2020.2969854>
- [22] Yang, M., Xu, J., Luo, K., & Zhang, Y. (2021). Sentiment analysis of Chinese text based on Elmo-RNN model. In *Journal of Physics: Conference Series* (Vol. 1748, No. 2, p. 022033). IOP Publishing.
- [23] Yanmei, L., & Yuda, C. (2015, December). Research on Chinese micro-blog sentiment analysis based on deep learning. In *2015 8th international symposium on computational intelligence and design (ISCID)* (Vol. 1, pp. 358-361). IEEE.
- [24] Zatarain Cabada, R., Barron Estrada, M., Oramas, R. (2018). Mining of educational opinions with deep learning. *Journal of Universal Computer Science*. 24. 1604-1626.
- [25] Zhou, J., Lu, Y., Dai, H. N., Wang, H., & Xiao, H. (2019). Sentiment analysis of Chinese microblog based on stacked bidirectional LSTM. *IEEE Access*, 7, 38856-38866.
- [26] Zhang, Y., Chen, M., Liu, L., & Wang, Y. (2017, June). An effective convolutional neural network model for Chinese sentiment analysis. In *AIP Conference Proceedings* (Vol. 1836, No. 1, p. 020085). AIP Publishing LLC.

## References

[27] Zhou, K., & Long, F. (2018, September). Sentiment analysis of text based on CNN and bi-directional LSTM model. In *2018 24th International Conference on Automation and Computing (ICAC)* (pp. 1-5). IEEE.

[28] Zhang, Z. (2020, March). Sentiment Analysis of Chinese Commodity Reviews Based on Deep Learning. In *International Conference on Modern Educational Technology and Innovation and Entrepreneurship (ICMETIE 2020)* (pp. 22-28). Atlantis Press.

## APPENDIX

### A.1 Weekly Report

## FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

<b>Trimester, Year:</b> Y4S2	<b>Study week no.:</b> 1
<b>Student Name &amp; ID:</b> Lee Hao Jie 1801544	
<b>Supervisor:</b> Dr Ramesh Kumar Ayyasamy	
<b>Project Title:</b> DEEP LEARNING MODEL FOR OPINION MINING	

#### 1. WORK DONE

The Deep learning model has completed the data pre-process and model design.

#### 2. WORK TO BE DONE

The report introduction.

#### 3. PROBLEMS ENCOUNTERED

None

#### 4. SELF EVALUATION OF THE PROGRESS

The FYP 2 schedule need to need to be follow so the project can be completed in time



-----  
--  
Supervisor's signature



-----  
Student's signature

## FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

<b>Trimester, Year:</b> Y4S2	<b>Study week no.:</b> 2
<b>Student Name &amp; ID:</b> Lee Hao Jie 1801544	
<b>Supervisor:</b> Dr Ramesh Kumar Ayyasamy	
<b>Project Title:</b> DEEP LEARNING MODEL FOR OPINION MINING	

### 1. WORK DONE

The Deep learning model has completed the data pre-process and model design and the Report Introduction chapter 1.

### 2. WORK TO BE DONE

The report Chapter 2 Literature Review and start the model train process.

### 3. PROBLEMS ENCOUNTERED

None

### 4. SELF EVALUATION OF THE PROGRESS

The FYP 2 schedule need to need to be follow so the project can be completed in time



-----  
Supervisor's signature



-----  
Student's signature

## FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

<b>Trimester, Year:</b> Y4S2	<b>Study week no.:</b> 3
<b>Student Name &amp; ID:</b> Lee Hao Jie 1801544	
<b>Supervisor:</b> Dr Ramesh Kumar Ayyasamy	
<b>Project Title:</b> DEEP LEARNING MODEL FOR OPINION MINING	

### 1. WORK DONE

The report Chapter 2 Literature Review and start the model train process.

### 2. WORK TO BE DONE

The Report Chapter 3 and 4 which is System Model and System Design.  
The model training has done and need to fine tune the model parameter.

### 3. PROBLEMS ENCOUNTERED

None

### 4. SELF EVALUATION OF THE PROGRESS

The FYP 2 schedule need to need to be follow so the project can be completed in time



-----  
Supervisor's signature



-----  
Student's signature

## FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

<b>Trimester, Year:</b> Y4S2	<b>Study week no.:</b> 4
<b>Student Name &amp; ID:</b> Lee Hao Jie 1801544	
<b>Supervisor:</b> Dr Ramesh Kumar Ayyasamy	
<b>Project Title:</b> DEEP LEARNING MODEL FOR OPINION MINING	

### 1. WORK DONE

The Report Chapter 3 and 4 which is System Model and System Design.  
The model fine tunes the model parameter process has done.

### 2. WORK TO BE DONE

The Report Chapter 5 and 6 which is the System Evaluation and Experiment.  
The model needs to conduct experiment and performance matrix graph.

### 3. PROBLEMS ENCOUNTERED

None

### 4. SELF EVALUATION OF THE PROGRESS

The FYP 2 schedule need to need to be follow so the project can be completed in time



-----  
Supervisor's signature



-----  
Student's signature



## FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

<b>Trimester, Year:</b> Y4S2	<b>Study week no.:</b> 5
<b>Student Name &amp; ID:</b> Lee Hao Jie 1801544	
<b>Supervisor:</b> Dr Ramesh Kumar Ayyasamy	
<b>Project Title:</b> DEEP LEARNING MODEL FOR OPINION MINING	

### 1. WORK DONE

The Report Chapter 5 and 6 which is the System Evaluation and Experiment. The model had conducted experiment with some sample and performance matrix graph.

### 2. WORK TO BE DONE

The Report Chapter 7 conclusion part and report plagiarism check in Turnitin.

### 3. PROBLEMS ENCOUNTERED

None

### 4. SELF EVALUATION OF THE PROGRESS

The FYP 2 schedule need to need to be follow so the project can be completed in time



-----  
Supervisor's signature



-----  
Student's signature

## FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

<b>Trimester, Year:</b> Y4S2	<b>Study week no.:</b> 6
<b>Student Name &amp; ID:</b> Lee Hao Jie 1801544	
<b>Supervisor:</b> Dr Ramesh Kumar Ayyasamy	
<b>Project Title:</b> DEEP LEARNING MODEL FOR OPINION MINING	

### 1. WORK DONE

All Report Work and model Work has been done

### 2. WORK TO BE DONE

None

### 3. PROBLEMS ENCOUNTERED

None

### 4. SELF EVALUATION OF THE PROGRESS

The FYP 2 schedule need to need to be follow so the project can be completed in time



-----  
Supervisor's signature



-----  
Student's signature

A.2 POSTER

# Deep Learning

## For Opinion Mining

By: Lee Hao Jie

**Chinese Opinion Mining**

Classify a review to positive and negative

**Performance**

95 % accuracy and F1-score on ChnSentiCorp

**Model Design**

It use Ernie as the pre-trained model and Fine tune it. The Fine Tune process can be done within an hour

Ernie model use different Mask strategy and Dialog language model has better understand on language context

Embedding

→

12\*Encoder

→

Pool Layer


**Input**

Room is small and dirty

**Output**


10%

→



90%

→



**Application**

- Hate Speech Detection
- Twitter topic analysis
- Social media monitoring

## PLAGIARISM CHECK RESULT

### Deep Learning for Opinion Mining

#### ORIGINALITY REPORT

**5%**

SIMILARITY INDEX

**3%**

INTERNET SOURCES

**3%**

PUBLICATIONS

**2%**

STUDENT PAPERS

#### PRIMARY SOURCES

**1**

[www.mdpi.com](http://www.mdpi.com)

Internet Source

<1 %

**2**

[fict.utar.edu.my](http://fict.utar.edu.my)

Internet Source

<1 %

**3**

Moreno-Ortiz, Antonio, and Javier Fernández-Cruz. "Identifying Polarity in Financial Texts for Sentiment Analysis: A Corpus-based Approach", *Procedia - Social and Behavioral Sciences*, 2015.

Publication

<1 %

**4**

[www.researchgate.net](http://www.researchgate.net)

Internet Source

<1 %

**5**

Walaa Medhat, Ahmed Hassan, Hoda Korashy. "Sentiment analysis algorithms and applications: A survey", *Ain Shams Engineering Journal*, 2014

Publication

<1 %

**6**

[repository.tudelft.nl](http://repository.tudelft.nl)

Internet Source

<1 %

7	Li Yang, Ying Li, Jin Wang, R. Simon Sherratt. "Sentiment Analysis for E-Commerce Product Reviews in Chinese Based on Sentiment Lexicon and Deep Learning", IEEE Access, 2020 Publication	<1%
8	"Natural Language Processing and Information Systems", Springer Science and Business Media LLC, 2019 Publication	<1%
9	Paddington Chiguvare, Christopher W Cleghorn. "Improving transformer model translation for low resource South African languages using BERT", 2021 IEEE Symposium Series on Computational Intelligence (SSCI), 2021 Publication	<1%
10	web.stanford.edu Internet Source	<1%
11	"Machine Learning and Knowledge Discovery in Databases", Springer Science and Business Media LLC, 2020 Publication	<1%
12	Submitted to University of Wales, Bangor Student Paper	<1%
13	Submitted to The Robert Gordon University Student Paper	<1%

14	Submitted to University of Exeter Student Paper	<1 %
15	tudr.thapar.edu:8080 Internet Source	<1 %
16	Submitted to University of Westminster Student Paper	<1 %
17	Chao-Wei Huang, Yun-Nung Chen. "Adapting Pretrained Transformer to Lattices for Spoken Language Understanding", 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), 2019 Publication	<1 %
18	Submitted to Universiti Tunku Abdul Rahman Student Paper	<1 %
19	scholar.sun.ac.za Internet Source	<1 %
20	Chieh-Yang Huang, Hanghang Tong, Jingrui He, Ross Maciejewski. "Location Prediction for Tweets", Frontiers in Big Data, 2019 Publication	<1 %
21	arxiv.org Internet Source	<1 %
22	"Twitter Spam Detection using Pre-trained Model", International Journal of Recent Technology and Engineering, 2019 Publication	<1 %

23	Manolis Maragoudakis. "A Review of Opinion Mining Methods for Analyzing Citizens' Contributions in Public Policy Debate", Lecture Notes in Computer Science, 2011 Publication	<1 %
24	www.ijert.org Internet Source	<1 %
25	"Sentiment Analysis and Opinion Mining", Encyclopedia of Machine Learning and Data Mining, 2016. Publication	<1 %
26	Rishika Garg, Mayank Singhal, Praveen Singh, Preeti Nagrath. "Chapter 71 Sentiment Analysis of Reviews Using Bi-LSTM Using a Fine-Grained Approach", Springer Science and Business Media LLC, 2023 Publication	<1 %
27	S. Thara, S. Sidharth. "Aspect based sentiment classification: Svd features", 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2017 Publication	<1 %
28	Submitted to University of Huddersfield Student Paper	<1 %
29	docplayer.net Internet Source	<1 %

30	eprints.uthm.edu.my Internet Source	<1 %
31	www.hindawi.com Internet Source	<1 %
32	"Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications", Springer Science and Business Media LLC, 2019 Publication	<1 %
33	Sarah Omar Alhumoud, Asma Ali Al Wazrah. "Arabic sentiment analysis using recurrent neural networks: a review", Artificial Intelligence Review, 2021 Publication	<1 %
34	Kansal, Hitesh, and Durga Toshniwal. "Aspect based Summarization of Context Dependent Opinion Words", Procedia Computer Science, 2014. Publication	<1 %

---

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		



<b>Universiti Tunku Abdul Rahman</b>			
<b>Form Title: Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)</b>			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1 of 1



**FACULTY OF INFORMATION AND COMMUNICATION  
TECHNOLOGY**

<b>Full Name(s) of Candidate(s)</b>	Lee Hao Jie
<b>ID Number(s)</b>	18ACB01544
<b>Programme / Course</b>	Bachelor of Computer Science (Honours)
<b>Title of Final Year Project</b>	DEEP LEARNING MODEL FOR OPINION MINING

<b>Similarity</b>	<b>Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)</b>
<b>Overall similarity index: <u>5</u> %</b>  <b>Similarity by source</b> Internet Sources: <u>3</u> % Publications: <u>3</u> % Student Papers: <u>2</u> %	
<b>Number of individual sources listed of more than 3% similarity: <u>0</u></b>	
<b>Parameters of originality required, and limits approved by UTAR are as Follows:</b> (i) Overall similarity index is 20% and below, and (ii) Matching of individual sources listed must be less than 3% each, and (iii) Matching texts in continuous block must not exceed 8 words <i>Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.</i>	

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

***Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.***

\_\_\_\_\_  
Signature of Supervisor

Name:  
Dr. Ramesh Kumar Ayyasamy

Date:  
30/11/2022

\_\_\_\_\_  
Signature of Co-Supervisor

Name:  
\_\_\_\_\_

Date:  
\_\_\_\_\_



## UNIVERSITI TUNKU ABDUL RAHMAN

### FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY (KAMPAR CAMPUS) CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	18ACB01544
Student Name	Lee Hao Jie
Supervisor Name	Dr Ramesh Kumar Ayyasamy

TICK (✓)	DOCUMENT ITEMS
	Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
	Front Plastic Cover (for hardcopy)
✓	Title Page
✓	Signed Report Status Declaration Form
✓	Signed FYP Thesis Submission Form
✓	Signed form of the Declaration of Originality
✓	Acknowledgement
✓	Abstract
✓	Table of Contents
✓	List of Figures (if applicable)
—	List of Tables (if applicable)
—	List of Symbols (if applicable)
✓	List of Abbreviations (if applicable)
✓	Chapters / Content
✓	Bibliography (or References)
✓	All references in bibliography are cited in the thesis, especially in the chapter of literature review
✓	Appendices (if applicable)
✓	Weekly Log
✓	Poster
✓	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)
✓	I agree 5 marks will be deducted due to incorrect format, declare wrongly the ticked of these items, and/or any dispute happening for these items in this report.

\*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report.

(Signature of Student)

Date: 01/12/2022