

國立臺灣師範大學  
資訊工程學系碩士論文

指導教授：侯文娟 博士

運用類神經網路方法分析  
基於面向的情感極性分類

Aspect-based Sentiment Polarity Classification  
using Neural Network Methods



研究生：王皓平 撰

中華民國 一百一十一年 六月

## 摘要

隨著時代以及科技技術的成長，人們不像過去一樣，需要查看報紙、購買雜誌、詢問左右鄰居的情報才能知道自己想要得知的資訊。在科技技術的成長下，不管是餐廳的評價、筆記型電腦的實用程度，大部分的人們都可以使用網際網路來查看是否有所想要的資訊。

本論文使用的資料集由 SemEval-2014 Task 4 官方所提供，並且含有四項子任務：(一) Aspect term extraction、(二) Aspect term polarity、(三) Aspect category detection、(四) Aspect category polarity，本論文進行第二項子任務研究，判斷出句子中的面向詞是正面、負面或中立，評估方式採用 Accuracy，並且與當年競賽結果相比較。

本論文實驗方法將資料先進行前處理並且轉成詞向量作為輸入的來源，以及將極性做情感標籤，並且使用 Bi-LSTM (Bi-directional Long Short-Term Memory)、Self-attention(自注意力機制)及使用 Two-level encoding 對資料進行訓練。

最後去比對每種不同模型的準確率，結果顯示 Two-level encoding 預測準確率餐廳達82%，筆記型電腦則達78%。

關鍵字:自然語言處理、雙向長短期記憶、自注意力機制、情感分析、編碼器、深度學習

## Abstract

With the development of the times and technology, people are not the same as in the past, that they need to check newspapers, buy magazines, and ask their neighbors for information to know what they want to know. With the development of technology, most people can use the Internet to check whether they have the information they want, no matter it is restaurant reviews or the practicality of laptops.

The dataset used in this study is officially provided by SemEval-2014 Task 4, which contains four subtasks: (1) Aspect term extraction, (2) Aspect term polarity, (3) Aspect category detection, (4) Aspect category polarity, This paper conducts the second sub-task study to judge whether the term words in the sentence are positive, negative or neutral.

The experimental method of this study pre-processes the data and converts it into the word vector as the source of input, and uses polarity as emotional label. Then Bi-LSTM (Bi-directional Long Short-Term Memory), Self-attention and Two-level encoding models are used to train the data.

Finally, to compare the accuracy of each different model, the results show that using the two-level encoding method, the prediction accuracy rate is 82% for restaurants, and 78% for laptops.

Keywords: Natural language processing, Bi-directional Long Short-Term Memory, Self-attention mechanism, sentiment analysis, encoder, deep learning

# 目錄

表目錄.....	v
圖目錄.....	vii
第一章 緒論.....	1
第一節 研究動機.....	1
第二節 研究目的.....	2
第三節 論文架構.....	3
第二章 文獻探討.....	4
第一節 自然語言處理工具包(NLTK).....	4
第二節 情感分析.....	5
第三節 長短期記憶模型.....	7
第四節 自注意力機制.....	8
第五節 條件隨機場.....	11
第三章 實驗方法與步驟.....	14
第一節 實驗資料.....	14
第二節 評估指標.....	15
第三節 實驗流程.....	16
第四節 資料整理.....	18
第五節 去除標點符號及停用詞.....	20
第六節 情感標籤.....	23
第七節 轉換為向量.....	24
第八節 雙向長短期記憶.....	28

第九節 Encoder-Decoder.....	29
第四章 實驗結果.....	32
第一節 開發環境.....	32
第二節 實驗結果.....	33
第三節 分析與討論.....	40
第五章 結論與未來展望.....	43
第一節 結論.....	43
第二節 未來研究方向.....	44
參考文獻.....	45



## 表目錄

表 2.1 語言處理任務對應的模組及功能.....	4
表 3.1 評估代號說明.....	16
表 3.2 餐廳評論測試集資料整理.....	18
表 3.3 筆記型電腦評論測試集資料整理.....	19
表 3.4 餐廳評論內容去除標點符號.....	20
表 3.5 筆記型電腦評論內容去除標點符號.....	21
表 3.6 NLTK 停用詞表 .....	21
表 3.7 餐廳評論去除標點符號及停用詞.....	22
表 3.8 筆記型電腦評論去除標點符號及停用詞.....	23
表 3.9 情感標籤極性對照圖.....	24
表 3.10 Encoder-Decoder 與 Two-level Encoder-Decoder 之 Training Loss 比較 .....	31
表 3.11 Encoder-Decoder 與 Two-level Encoder-Decoder 餐廳評論準確度比較.....	31
表 3.12 Encoder-Decoder 與 Two-level Encoder-Decoder 筆記型電腦評論準確度比較..	31
表 4.1 硬體開發環境.....	32
表 4.2 軟體開發環境.....	32
表 4.3 使用 Word2vec 詞嵌入前處理五種不同模型準確度(餐廳).....	36
表 4.4 使用 Word2vec 詞嵌入前處理五種不同模型準確度(筆記型電腦).....	36
表 4.5 使用 Glove 詞嵌入前處理五種不同模型準確度(餐廳) .....	37
表 4.6 使用 Glove 詞嵌入前處理五種不同模型準確度(筆記型電腦) .....	37
表 4.7 使用 Word2vec 詞嵌入前處理五種不同模型訓練時間 (餐廳).....	38
表 4.8 使用 Word2vec 詞嵌入前處理五種不同模型訓練時間(筆記型電腦).....	38

表4.9 使用 Glove 詞嵌入前處理五種不同模型訓練時間(餐廳).....	39
表4.10 使用 Glove 詞嵌入前處理五種不同模型訓練時間(筆記型電腦).....	39
表 4.11 SemEval-2014 餐廳評論前十名團隊與本實驗之準確度.....	41
表 4.12 SemEval-2014 筆記型電腦評論前十名團隊與本實驗之準確度.....	41



## 圖目錄

圖 2.1 基本情感分析極性分類流程圖.....	6
圖 2.2 多種情感面向分析極性分類流程圖.....	6
圖 2.3 長短期記憶模型內部結構.....	7
圖 2.4 自注意力機制取得 $\alpha$ .....	9
圖 2.5 自注意力機制取得 $b1$ .....	10
圖 2.6 隱形馬爾科夫模型架構.....	12
圖 2.7 條件隨機場架構(一).....	12
圖 2.8 條件隨機場架構(二).....	13
圖 3.1 SemEval-2014 Task 4 餐廳評論資料集 A 範例.....	14
圖 3.2 SemEval-2014 Task 4 餐廳評論資料集 B 範例.....	15
圖 3.3 SemEval-2014 Task 4 筆記型電腦評論資料集 A 範例.....	15
圖 3.4 SemEval-2014 Task 4 筆記型電腦評論資料集 B 範例.....	15
圖 3.5 研究架構整體流程圖.....	17
圖 3.6 CBOW 與 Skip-gram 模型架構圖 .....	26
圖 3.7 詞彙轉換詞向量流程圖.....	27
圖 3.8 雙向長短期記憶模型架構.....	28
圖 3.9 Encoder-Decoder 架構.....	29
圖 3.10 One-level Encoder-Decoder 模型架構.....	30
圖 4.1 LSTM 模型架構.....	33
圖 4.2 Bi-LSTM 模型架構.....	34
圖 4.3 Bi-LSTM + Self-Attention 模型架構.....	34

圖 4.4 Bi-LSTM + Self-Attention + CRF 模型架構 .....35

圖 4.5 Two-Level Encoding 模型架構 .....35



# 第一章 緒論

## 第一節 研究動機

隨著時代以及科技技術的成長，人們不像過去一樣，需要查看報紙、購買雜誌、詢問左右鄰居的情報才能知道自己想要得知的資訊。在科技技術的成長下，不管是餐廳的評價、筆記型電腦的實用程度，大部分的人們都可以使用網際網路來查看是否有所想要的資訊。但是文字的表達與語言的表達所呈現的方式，兩者都有所不同，在文字中很難去感受出文字所真正想表達的情感。像是在許多餐廳的評價中，每位顧客想知道的資訊都有所不同，有些人想要知道食物相關的評價或者服務方面的評價，但是一些評價的句子可能一起含有正面及負面的意思，很容易使人搞混這個評價想要表達的含意。因此，如何從這個句子中找出想要表達的情感，是非常重要的以及具有挑戰性的議題。

在自然語言處理(Natural Language Processing, NLP)中應用了許多生活裡所需要用到的任務，像是分辨真假新聞、句子情感分類任務、語言翻譯等等。其中句子情感分類任務，也包含評價的部份，現今外送平台越來越多，餐廳的評價也是許多人用餐的一個選擇。往往評論不只對單一面向詞做描述，而是對多個面向詞做不同的描述，為了解決上述問題，本研究應用 Aspect-based sentiment analysis(ABSA)，分析不同面向詞所代表的情感，而 Gao 等人(Gao, Z., Feng, A., Song, X., & Wu, X., 2019)也曾分析句子中不同面向，對於不同面向詞進行情感極

性的分類，讓許多人能夠更方便的知道該句中的面向所代表的情感傾向，也是本論文研究動機之一。

## 第二節 研究目的

計算語言協會(Association for computational linguistics, ACL)在2014年舉辦國際語意評估研討會，在這場研討會中提出了十種自然語言處理的任務，其中第四種任務為基於面向情感分析(Aspect Based Sentiment Analysis)，在 SemEval-2014 Task 4 (Pontiki, M., Galanis, D., Pavlopoulos, J., Papageorgiou, H., Androutsopoulos, I., & Manandhar, S. 2014)中總共有四個子任務：分別為(一) Aspect term extraction、(二) Aspect term polarity、(三) Aspect category detection、(四) Aspect category polarity，本研究使用 SemEval-2014 Task 4所提供之資料集，進行第二個子任務，在第二個子任務中我們將判斷這個句子中的面向詞是正面、負面或是中立，如例句 1 所示，並且分析三種極性的準確性。

例句 1：While it was large and a bit noisy, the drinks were fantastic, and the food was superb.

與一般直接從整個句子來判斷極性並不相同，在本研究中我們會根據每個面向來判斷極性，在例句 1 中 drinks 所代表的極性為正面，而 food 也是為正面，但如果用整個句子去分析，它可能會因為 noisy 而判斷成負面情感，這樣就導致判斷錯誤，也無法得知這家餐廳到底哪裡不好，如此可能會造成顧客選擇上

的影響，本研究分析多種面向之情感，對於每個面向給出三種極性的分數，以便給予顧客更好的建議。

### 第三節 論文架構

本論文的架構如下，第一章為緒論，說明研究動機以及研究目的；第二章為文獻探討，說明本研究中所使用到的方法或者相關文獻；第三章為實驗方法與步驟；第四章為實驗結果，說明本實驗之結果，與當年比賽團隊的實驗結果之比較；第五章為結論與未來展望。



## 第二章 文獻探討

### 第一節 自然語言處理工具包(NLTK)

NLTK<sup>1</sup>是在自然語言處理中，處理英文以及其他西方語言句子時常常需要使用到的工具，全名為 Natural Language Toolkit，由 Steven Bird、Edward Loper 和 Ewan Klein (Wagner, W. 2010)在 Python 上所開發的模組，並且以簡單性、一致性、可擴充套件性以及模組化為設計目標，它提供了許多在文本前處理時常用到的項目如表 2.1 所示。

表 2.1：語言處理任務對應的模組及功能

(表格參考自：<https://www.nltk.org/book/>)

語言處理任務	NLTK 模組	功能說明
Accessing corpora	corpus	standardized interfaces to corpora and lexicons
String processing	tokenize, stem	tokenizers, sentence tokenizers, stemmers
Collocation discovery	collocations	t-test, chi-squared, point-wise mutual information
Part-of-speech tagging	tag	n-gram, backoff, Brill, HMM, TnT
Machine learning	classify, cluster, tbl	decision tree, maximum entropy, naive Bayes, EM, k-means

<sup>1</sup> <https://www.nltk.org/book/>

Chunking	chunk	regular expression, n-gram, named-entity
Parsing	parse, ccg	chart, feature-based, unification, probabilistic, dependency
Semantic interpretation	sem, inference	lambda calculus, first-order logic, model checking
Evaluation metrics	metrics	precision, recall, agreement coefficients
Probability and estimation	probability	frequency distributions, smoothed probability distributions
Applications	app, chat	graphical concordancer, parsers, WordNet browser, chatbots
Linguistic fieldwork	toolbox	manipulate data in SIL Toolbox format

本篇論文所使用到的模組為 tokenize 以及 corpus，分別讓句子去做斷詞以及獲取 NLTK 裡的語料庫，進行 Stop words removal，去除句子中非必要的詞。

## 第二節 情感分析

情感分析又稱為意見挖掘，是屬於自然語言處理中研究領域之一，許多研究者對此領域也非常有興趣。文本情感分析則是根據文本中的句子判斷是屬於正面、負面或中性，在圖2.1中，若輸入例句：「這間餐廳的服務很差。」，則會分類成負面，這句話只包含一種情感面向分析，若有兩種或多種情感面向時，例如：「這家餐廳的飲料很好喝，但是服務很差，而且價格也很昂貴」，在這個句子中有三種不同的情感面向，分別為「飲料」、「服務」以及「價格」，則會根

據更為複雜的處理方式來對這句話進行情感分析，如圖2.2所示，也有部分研究是使用高興、傷心、生氣、害怕等等，較為更複雜的情緒狀態來分析這段句子是屬於哪一方面的情緒。

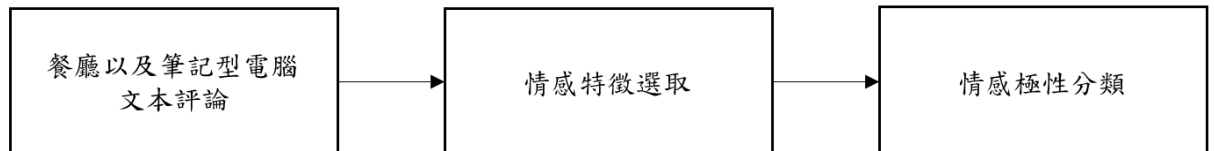


圖 2.1：基本情感分析極性分類流程圖

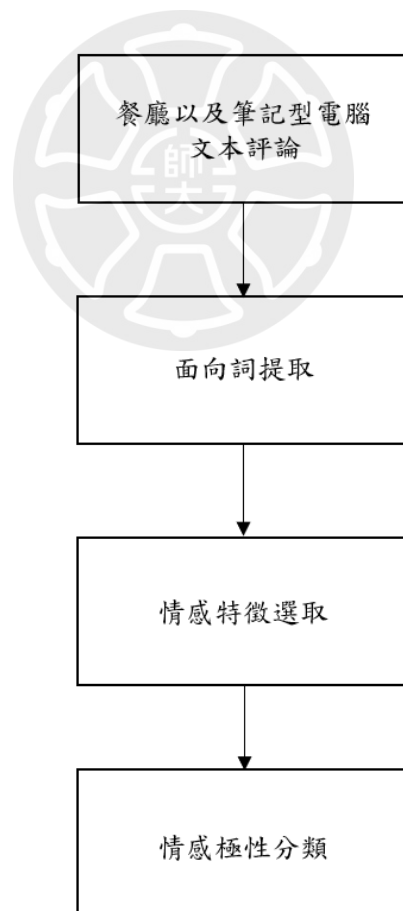


圖 2.2：多種情感面向分析極性分類流程圖

### 第三節 長短期記憶模型

長短期記憶(Long Short-term Memory, LSTM)(Hochreiter, S., & Schmidhuber, J. 1997)，改善許多遞迴神經網路(Recurrent Neural Network, RNN)(Zaremba, W., Sutskever, I., & Vinyals, O. 2014)所存在的問題，像是梯度消失及梯度爆炸的問題，因此長短期記憶導入三種控制記憶機制，分別為輸入閘(Input Gate)、遺忘閘(Forget Gate)、輸出閘(Output Gate)。輸入閘決定輸入的資料是否需要進入神經單元，遺忘閘決定這次進來的資料哪一些是需要被記憶起來的，哪一些是要進行遺忘的動作，輸出閘決定是否要輸出當下輸入的資料，如圖 2.3 所示。

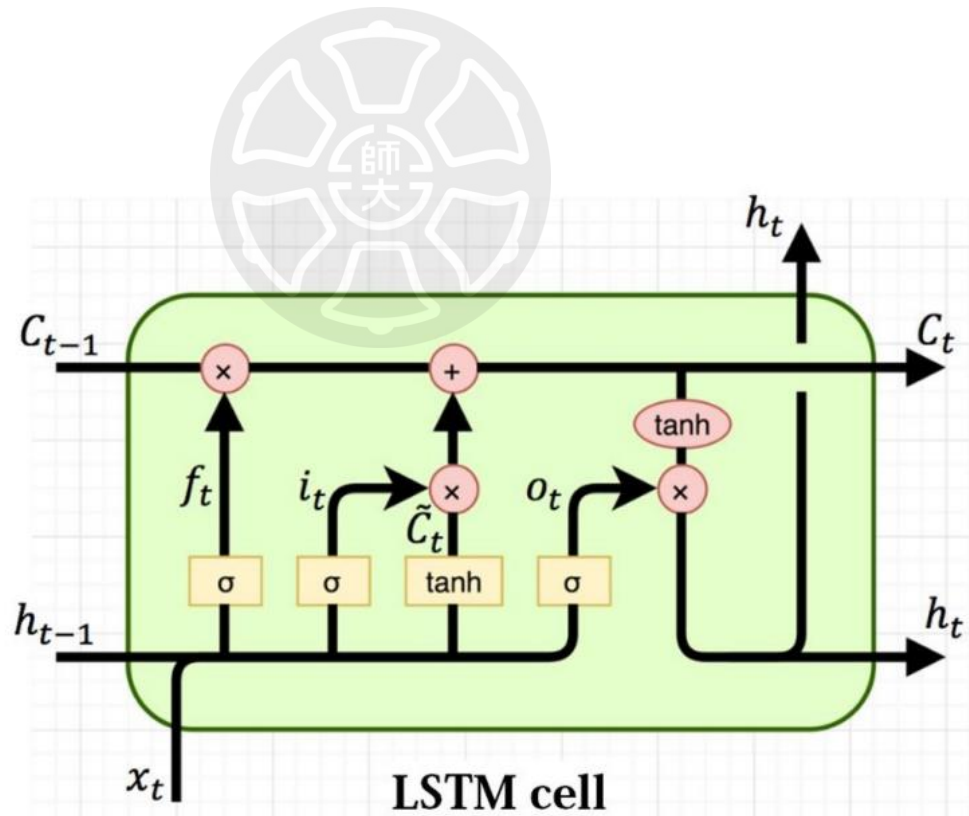


圖 2.3：長短期記憶模型內部結構

(圖片參考自：Varsamopoulos, S., Bertels, K., & Almudever, C. G. 2018)

其中 $x_t$ 為當前輸入資料， $f_t$ 為遺忘閘， $i_t$ 為輸入閘， $o_t$ 為輸出閘， $h_t$ 為各個隱藏層的輸出向量， $h_{t-1}$ 為前一個時序中的輸出， $C_t$ 為當下記憶體儲存內容， $C_{t-1}$ 為上一個記憶體儲存的内容， $\tilde{C}_t$ 為記憶體更新， $\sigma$ 為 S 形函數。當遺忘閘、輸出閘、輸入閘與輸入的資料做運算後，經過 S 形函數判斷是否打開，得出1則打開，得出0則關閉不允許輸入， $\tanh$  為雙取正切函數，當資料輸入或輸出時，會經過  $\tanh$  函數將數值控制在-1到1之間。

越來越多研究者也致力於發展更多長短期記憶模型的衍生並且使用於研究中，像是 Xie 等人(Xie, J., Chen, B., Gu, X., Liang, F., & Xu, X. 2019)，結合了雙向長短期記憶模型，讓特徵能夠判斷的更準確，Tang 等人(Tang, D., Qin, B., Feng, X., & Liu, T. 2015)，結合目標相關長短期記憶模型(Target Dependent Long Short-term Memory, TD-LSTM)使面向相關語義資訊更為豐富，判斷極性更為準確。

#### 第四節 自注意力機制

自注意力機制(Self-Attention)由 Vaswani 等學者於2017年提出(Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., & Polosukhin, I. 2017)許多人也使用此方法，應用於不同的任務，像是 Nguyen 等人(Nguyen, H. T., & Le Nguyen, M. 2018)用於分類任務，此方法不僅可以處理序列型資料也處理平行化運算，輸入的序列順序也不影響運作的結果，對於本研究的需求效果是非常好的。運作方式為它將輸入的向量資訊，同時考慮整個句子的向量，輸出成相同長度的向量資訊，後續再接到不同的神經網路中決定出它是哪一種類別的形式，由於需要考慮整個句子，決定出每一個向量與另一個向量有什麼關聯，所

以會有一個數值來計算，這裡稱為  $\alpha$ ，讓每一個向量去乘上不同的矩陣，得出向量後再做內積(inner product)，就可以得到  $\alpha$  這個數值，上述運作內容如圖2.4所示。

圖2.4中  $x_1$ 、 $x_2$  代表輸入向量， $w^q$ 、 $w^k$  代表兩種不同的矩陣， $q_1$  代表  $x_1$  及  $w^q$  相乘後的結果， $k_2$  代表  $x_2$  及  $w^k$  相乘後的結果， $\alpha_{1,2}$  代表  $q_1$  及  $k_2$  做內積的結果。

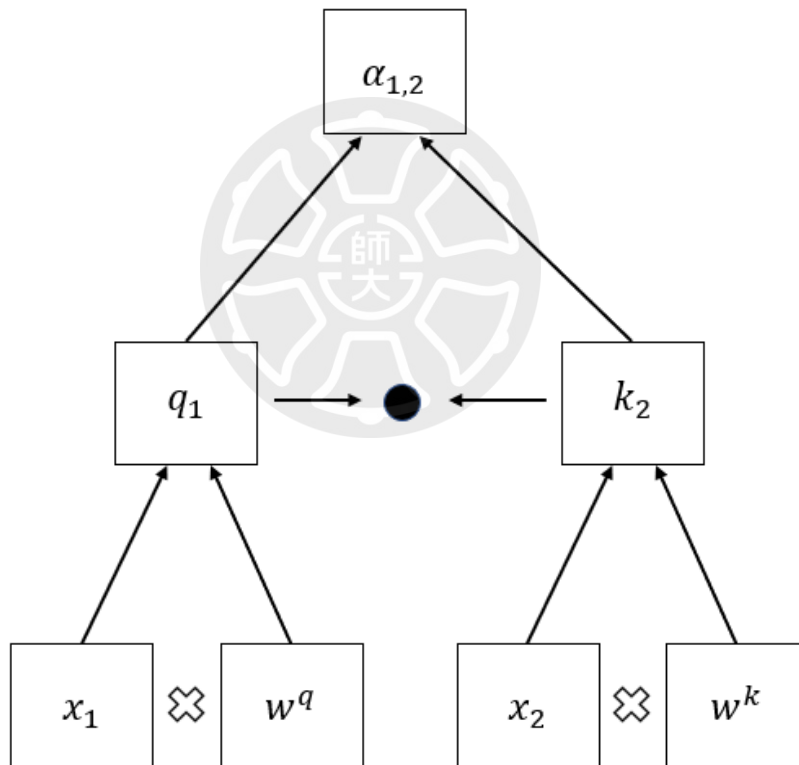


圖 2.4：自注意力機制取得  $\alpha$

從圖2.4得到  $\alpha$  後，經由不同的激勵函數，如：Softmax、ReLU 等，得到  $\alpha'$ ，得到  $\alpha'$  後，需要去找這個序列中重要的資訊，如圖2.5，以  $\alpha_{1,2}$  為例，首先將  $x_2$  乘

上  $w^v$  矩陣得到  $v_2$ ，得到  $v_2$  後再與  $\alpha'_{1,2}$  相乘，得出來的值再與其他  $\alpha'$  與  $v$  相乘的結果做相加得到  $b_1$ ， $b_1$  公式如下，假設  $\alpha'_{1,2}$  分數最大，則  $b_1$  值會越接近  $v_2$ 。

$$b_1 = \sum_i \alpha'_{1,i} v_i \quad (1)$$

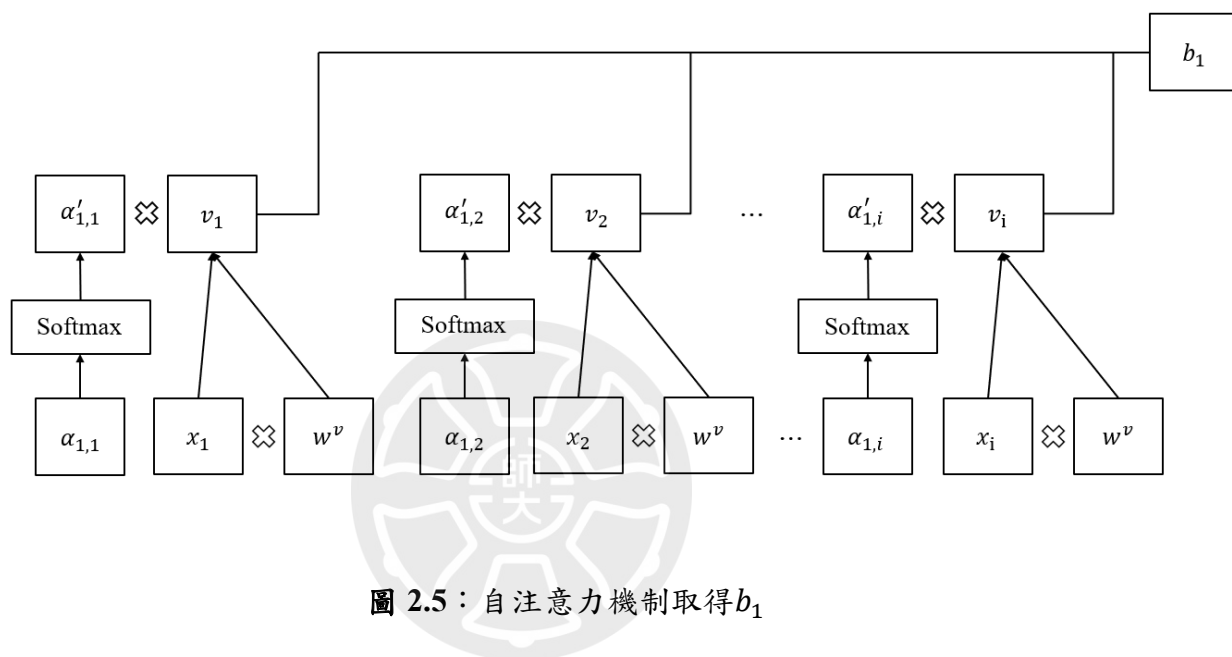


圖 2.5：自注意力機制取得  $b_1$

經由上述內容可以得到 Q(query)、K(key)、V(value)，公式如下：

$$qi = w^q x_i \quad (2)$$

$$ki = w^k x_i \quad (3)$$

$$vi = w^v x_i \quad (4)$$

其中  $qi$  代表 Q， $ki$  代表 K， $vi$  代表 V， $w^q$ 、 $w^k$ 、 $w^v$  代表三種不同的矩陣， $x_i$  代表輸入向量。

得到 Q、K、V 後，就能得出 Attention Matrix，詳細公式如下：

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) V \quad (5)$$

其中 $d_k$ 為 key 的維度，在 Vaswani 等人(Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., & Polosukhin, I. 2017)論文中，激勵函數使用 Softmax 進行歸一化計算權重，並除以 $d_k$ 目的為了防止內積運算結果過大。

## 第五節 條件隨機場

條件隨機場(Conditional Random Fields)，簡稱 CRFs (Lafferty, J., McCallum, A., & Pereira, F. C. 2001)，由 Lafferty 等人於2001年提出，CRFs 常常被應用在自然語言處理應用中，像是詞性標記、分析序列資料、斷詞、命名實體識別、翻譯等。

隱形馬爾科夫模型(Hidden Markov Models)，簡稱 HMMs，模型架構如圖2.6所示，其中 X 代表可觀察的輸出，Y 代表隱含狀態，是觀察者無法得知的變數。在詞性標記中，詞彙中的詞性非常重要，而在隱形馬爾科夫模型中，如圖2.6可以看出隱形馬爾科夫模型在某一個觀察節點時僅依賴對應的狀態節點，向右箭頭代表 Y 目前狀態的值與上一個 Y 狀態的值有所相關，而向下箭頭代表，如圖2.6，每個 $X_i$ 只和 $Y_i$ 有所相關，每個 $X_{i-1}$ 只和 $Y_{i-1}$ 有所相關，以此類推，隱形馬爾科夫模型缺乏了獨立性的假設，導致無法考慮到上下文的特徵，而條件隨機場可以補足這個問題。

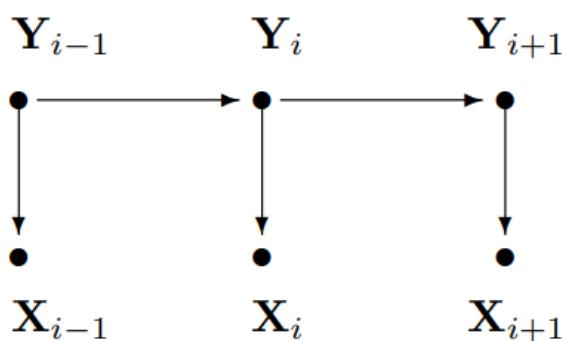


圖 2.6：隱形馬爾科夫模型架構

(圖片參考自：Lafferty, J., McCallum, A., & Pereira, F. C. 2001)

不同於隱形馬爾科夫模型的是條件隨機場為無向圖之模型，在條件隨機場中，對於輸出狀態序列  $Y$ ， $Y$  的值都與相鄰的  $Y$  有相互關係，而每個狀態序列  $Y$  都與輸入觀察值序列  $X$  有相互關係，兩種模型架構如圖 2.7、圖 2.8 所示。

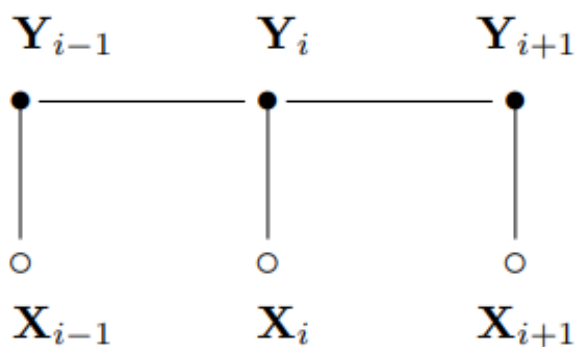


圖 2.7：條件隨機場架構(一)

(圖片參考自：Lafferty, J., McCallum, A., & Pereira, F. C. 2001)

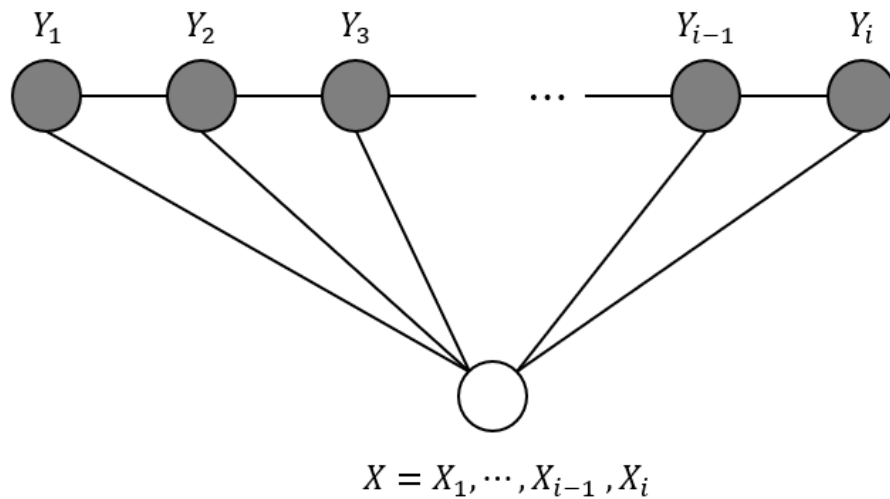


圖 2.8：條件隨機場架構(二)  
 (圖片參考自：Wallach, H. M. 2004)

條件隨機場模型公式如下：

$$P(Y|X) = \frac{1}{Z(x)} \exp \left\{ \sum_{i,k} \lambda_k f_k(y_{i-1}, y_i, x, t) \right\} \quad (6)$$

$$\frac{1}{Z(x)} = \sum_y \exp \left\{ \sum_{i,k} \lambda_k f_k(y_{i-1}, y_i, x, t) \right\} \quad (7)$$

其中 $Z(x)$ 為歸一化因子，在給定的輸入上讓狀態序列機率和為1， $f_k$ 為特徵集合， $t$ 代表時間標記上特徵函數， $\lambda_k$ 為每個 $f_k$ 經由訓練過後所對應的權重值， $x$ 代表輸入觀察值序列， $y_i$ 、 $y_{i-1}$ 代表輸出狀態序列。

## 第三章 實驗方法與步驟

### 第一節 實驗資料

本論文使用的資料集為 SemEval-2014 Task 4 主辦單位所提供的資料集，內容提供了許多餐廳以及筆記型電腦相關方面評論，並且提供兩種不同的資料集分別處理不同子任務，圖 3.1、圖 3.2 為餐廳評論，而在筆記型電腦也提供兩種不同的資料集如圖 3.3、圖 3.4。在圖 3.1、圖 3.3 中官方資料集只有給出一段句子，較適合處理子任務一，將句子中的面向詞提取出來。在圖 3.2 中官方資料集已經將每個句子的面向詞以及面向詞的類別都已經標示出來，並且標記每個面向詞以及面向詞類別在每個句子中所代表的極性，適合處理子任務二、三、四。在圖 3.4 中官方資料集只有標記出每個句子中的面向詞以及面向詞所代表的極性，只適合處理子任務二。而本研究是處理子任務二，判斷句子中面向詞的三種極性，所以較適合用資料集 B，在餐廳資料集 B 提供 3608 筆評論，而在筆記型電腦資料集 B 提供 2328 筆評論。

```
<sentence id="32897564#894393#2">  
  <text>The bread is top notch as well.</text>  
</sentence>
```

圖 3.1：SemEval-2014 Task 4 餐廳評論資料集 A 範例

```

<sentences>
  <sentence id="3121">
    <text>But the staff was so horrible to us.</text>
    <aspectTerms>
      <aspectTerm term="staff" polarity="negative" from="8" to="13"/>
    </aspectTerms>
    <aspectCategories>
      <aspectCategory category="service" polarity="negative"/>
    </aspectCategories>
  </sentence>

```

圖 3.2 : SemEval-2014 Task 4 餐廳評論資料集 B 範例

```

<sentence id="1615">
  <text>I bought this laptop on Saturday and am completely in love with it!</text>
</sentence>
<sentence id="1670">
  <text>If you don't like fingerprints, this might not be the laptop for you.</text>
</sentence>

```

圖 3.3 : SemEval-2014 Task 4 筆記型電腦評論資料集 A 範例

```

<sentences>
  <sentence id="2339">
    <text>I charge it at night and skip taking the cord with me because of the good battery life.</text>
    <aspectTerms>
      <aspectTerm term="cord" polarity="neutral" from="41" to="45"/>
      <aspectTerm term="battery life" polarity="positive" from="74" to="86"/>
    </aspectTerms>
  </sentence>

```

圖 3.4 : SemEval-2014 Task 4 筆記型電腦評論資料集 B 範例

## 第二節 評估指標

在機器學習分類模型中有許多評估方式，例如 Accuracy、Precision、Recall、F1-score，因為 SemEval-2014 Task 4 比賽中採取 Accuracy 當作評估指標，所以本論文也使用 Accuracy 當作本實驗的評估方式，依據 Accuracy 之定義，模型的準確率公式如下：

$$N = TP1+TP2+TP3+FP1+FP2+FP3 \quad (8)$$

$$\text{Accuracy} = \frac{TP1+TP2+TP3}{N} \quad (9)$$

其中 N 代表測試評論資料所有的樣本總數，TP1 代表預測正面正確樣本數，TP2 代表預測負面正確樣本數，TP3 代表預測中性正確樣本數，如表 3.1 所示

表 3.1：評估代號說明

代號	說明
TP1	預測正面正確樣本數
TP2	預測負面正確樣本數
TP3	預測中性正確樣本數
FP1	預測正面錯誤樣本數
FP2	預測負面錯誤樣本數
FP3	預測中性錯誤樣本數
N	測試評論資料樣本總數

### 第三節 實驗流程

由於圖 3.1、圖 3.3 資料集較適合子任務 1 做面向詞的提取，所以本篇論文使用圖 3.2、圖 3.4 來當作此實驗的資料集。另外官方的檔案是 Xml 形式，所以本實驗需要先將圖 3.2、圖 3.4 整理成表格的形式，以便進行前處理的工作，再將評論裡的句子做斷詞及刪除標點符號和停用詞去除 Stop words removal，再來將評論

以及面向詞轉換成詞向量，讓資料能夠被讀取。本研究採 Word2Vec 與 Glove 來完成轉化詞向量的工作，接著將極性做情感標籤使用 One-hot Encoding，給出三種

極性的三維向量，正面代表[1,0,0]，負面代表[0,1,0]，中性代表[0,0,1]。再來將處理好的資料做 Embedding，將資料轉換成可以讓神經模型可以輸入的形式，並且加上自注意力機制、雙向長短期記憶以及 Two-Level Encoding 決定每個向量與另一個向量之間有無關聯，最後將每一個面向詞去判斷三種極性，模型整體架構如圖3.5所示。

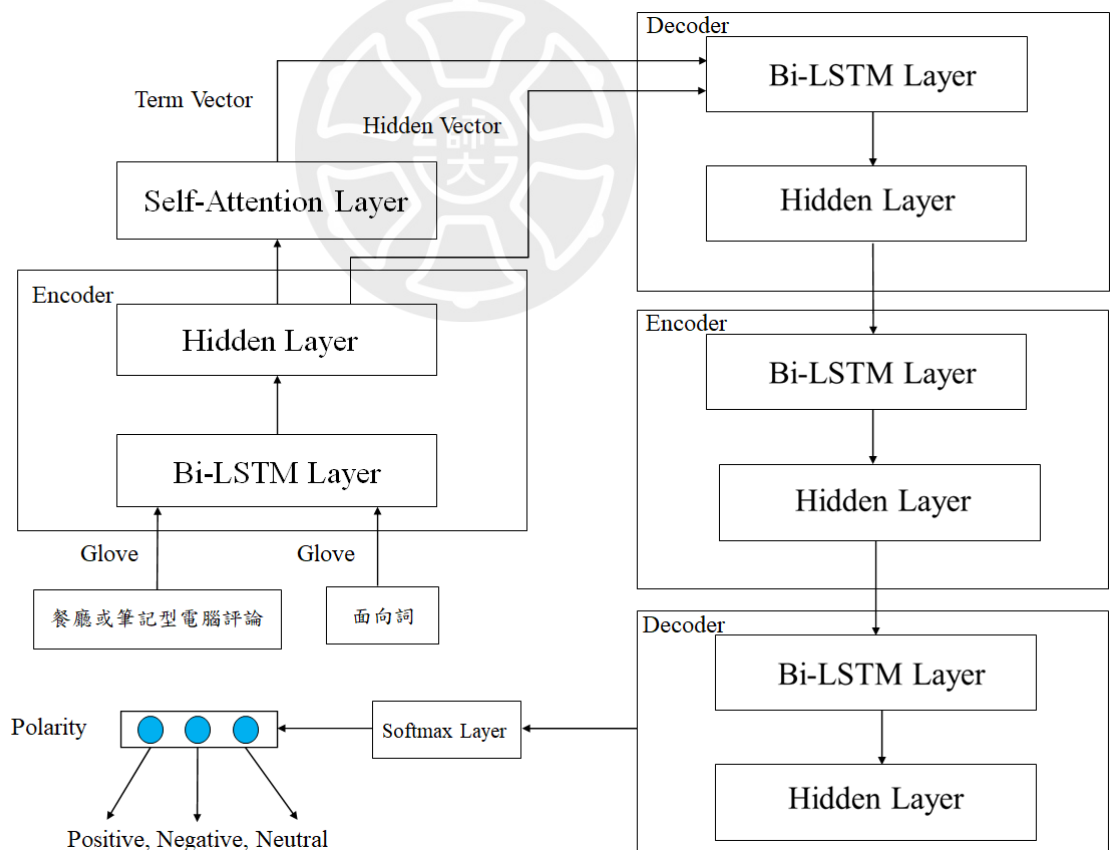


圖 3.5：研究架構整體流程圖

## 第四節 資料整理

在圖3.2、圖3.4中，官方已經將每一個句子的面向詞以及面向詞的極性都標示出來，我們只需將資料整理成表格的形式，讓電腦能夠讀出官方所提供的資料集每一個欄位都代表著什麼內容。在表3.2及表3.3中 id 為測試集資料的筆數編號，由於筆數過多，所以這裡只列出4或5筆的句子來呈現，text 為官方所提供的評論，term 為面向詞，由於一個評論可能有多種的面向詞，如表3.2中所示，第二到第四筆為一樣的評論，但 term 欄位的內容都不同，polarity 為每一個評論中的面向詞所代表的極性。

表 3.2：餐廳評論測試集資料整理

id	text	term	polarity
0	But the staff was so horrible to us.	staff	negative
1	To be completely fair, the only redeeming factor was the food, which was above average, but couldn't make up for all the other deficiencies of Teodora.	food	positive
2	The food is uniformly exceptional, with a very capable kitchen which will proudly whip up whatever you feel like eating, whether it's on the menu or not.	food	positive
3	The food is uniformly exceptional, with a very capable kitchen which will proudly whip up	kitchen	positive

	whatever you feel like eating, whether it's on the menu or not.		
4	The food is uniformly exceptional, with a very capable kitchen which will proudly whip up whatever you feel like eating, whether it's on the menu or not.	menu	neutral

表 3.3：筆記型電腦評論測試集資料整理

id	text	term	polarity
0	I charge it at night and skip taking the cord with me because of the good battery life.	cord	neutral
1	I charge it at night and skip taking the cord with me because of the good battery life	battery life	positive
2	it is of high quality, has a killer GUI, is extremely stable, is highly expandable, is bundled with lots of very good applications, is easy to use, and is absolutely gorgeous.	quality	positive
3	it is of high quality, has a killer GUI, is extremely stable, is highly expandable, is bundled with lots of very good applications, is easy to use, and is absolutely gorgeous.	GUI	positive
4	it is of high quality, has a killer GUI, is extremely stable, is highly expandable, is bundled with lots of very good applications, is easy to use, and is absolutely gorgeous.	applications	positive

5	it is of high quality, has a killer GUI, is extremely stable, is highly expandable, is bundled with lots of very good applications, is easy to use, and is absolutely gorgeous.	use	positive
---	---	-----	----------

## 第五節 去除標點符號及停用詞

因為在評論中有許多標點符號以及無意義的單詞，這樣的情況會讓模型在訓練的時候，造成訓練到無意義的單詞，而導致準確率下降，而在 Python 函數中可以使用正規表達式(re)來刪除標點符號及數字的情況，本實驗所用到的函式為：「`text = re.sub("[^A-Z^a-z^0-9]", " ", text)`」，能有效的刪除標點符號及數字，如表3.4與表3.5所示：

表 3.4：餐廳評論內容去除標點符號

原始餐廳評論內容	Ok, so the servers wander around a little clueless, but there's more than enough servers for the crowd they get -- it's fine, you just have to make a small effort to get their attention.
餐廳評論內容去除標點符號	Ok so the servers wander around a little clueless but there s more than enough servers for the crowd they get it s fine you just have to make a small effort to get their attention

表 3.5：筆記型電腦評論內容去除標點符號

原始筆記型電腦評論內容	it is of high quality, has a killer GUI, is extremely stable, is highly expandable, is bundled with lots of very good applications, is easy to use, and is absolutely gorgeous.
筆記型電腦評論內容 去除標點符號	it is of high quality has a killer GUI is extremely stable is highly expandable is bundled with lots of very good applications is easy to use and is absolutely gorgeous

在自然語言處理中，為了節省儲存空間以及提高檢索的效率，會將一些字或詞自動過濾，比方說文字出現頻率較高或者沒有統計意義的詞，這些字或詞稱為 Stop Words(停用詞)，停用詞沒有一定的格式，都是由人工所輸入，每個人所認為的停用詞都不一定相同。

本研究使用 Python nltk 之停用詞，停用詞表也會定期的進行更新，停用詞表如表3.6所示，共有178個停用詞，最後呈現出評論去除標點符號及停用詞的結果，如表3.7及表3.8所示。

表 3.6：NLTK 停用詞表

I	Me	My	Myself	We	Our	Ours
Ourselves	You	You're	You've	You'll	You'd	Your
Yours	Yourself	Yourselves	He	Him	His	Himself
She	She's	Her	Hers	Herself	It	It's
Its	Itself	They	Them	Theirs	Themselves	What
Which	Who	Whom	This	That	That'll	These
Those	Am	Is	Are	Was	Were	Be

Been	Being	Have	Has	Had	Having	Do
Does	Did	Doing	A	An	The	And
But	If	Or	Because	As	Until	While
Of	At	By	For	With	About	Against
Between	Into	Through	During	Before	After	Above
Below	To	Form	Up	Down	In	Out
On	Off	Over	Under	Again	Further	Then
Once	Here	There	When	Where	Why	How
All	Any	Both	Each	Few	More	Most
Other	Some	Such	No	Nor	Not	Only
Own	Same	So	Than	Too	Very	S
T	Can	Will	Just	Don	Don't	should
Should've	Now	D	ll	M	O	Re
Ve	Y	Ain	Aren	Aren't	Couldn	Couldn't
Didn	Didn't	Doesn	Doesn't	Hadn	Hadn't	Hasn
Hasn't	Haven	Haven't	Isn	Isn't	Ma	Mightn
Mightn't	Mustn	Mustn't	Needn	Needn't	Shan	Shan't
Shouldn	Shouldn't	Wasn	Wasn't	Weren	Weren't	Won
Won't	Wouldn	Wouldn't				

表 3.7：餐廳評論去除標點符號及停用詞

原始餐廳評論內容	Ok, so the servers wander around a little clueless, but there's more than enough servers for the crowd they get -- it's fine, you just have to make a small effort to get their attention.
----------	--

去除標點符號及停用詞餐廳評論內容	Ok servers wander around a little clueless but enough servers crowd get fine make a small effort to get their attention
------------------	---

表 3.8：筆記型電腦評論去除標點符號及停用詞

原始筆記型電腦評論內容	it is of high quality, has a killer GUI, is extremely stable, is highly expandable, is bundled with lots of very good applications, is easy to use, and is absolutely gorgeous.
去除標點符號及停用詞筆記型電腦評論內容	high quality killer GUI extremely stable highly expandable bundled lots good applications easy use absolutely gorgeous

## 第六節 情感標籤



在資料集中情感極性並未給一個數值，所以首先需要將三種不同的極性給定不同的數字，接著轉換成不同向量，幫助模型去判斷及預測結果。情感標籤對照表如表3.9所示，假設預測分析結果為[0.2,0.7,0.1]，則可以發現負面情感在預測的結果是比例占最高的，則此分析預測的結果為負面評論。在知道分析預測的結果及正確的結果後，透過損失函數計算的差距，模型則會自動修正學習的方向，讓下次的預測能夠讓分析預測結果及正確的結果差距更小。

表 3.9：情感標籤極性對照表

情感極性	數字形式	向量
正面	+1	[1,0,0]
負面	-1	[0,1,0]
中性	+0	[0,0,1]

## 第七節 轉換為向量

詞嵌入(Word embedding)在自然語言處理中非常廣泛使用的一項技術，主要目的是為了將輸入的一整句句子轉換成向量，並將轉換後的向量，送至每個不同模型的裡面去做後續的應用。

詞嵌入的做法有很多種，在早期的做法有詞袋模型(Bag-of-Word)( Zhang, Y., Jin, R., & Zhou, Z. H. 2010)以及獨熱編碼(One-hot encoding)，詞袋模型之做法為將一整個句子拆成多個個詞彙，並且想像是放在一個袋子裡，袋子裡面的詞彙沒有順序關係，都是獨立的單位，這種做法簡化了許多文件自動處理的計算。在獨熱編碼中，將一個句子轉換成向量形式表達，它會根據詞彙的數量來建立不同維度的向量，在這個向量表示中，其內容只有一個維度為1，其餘則為0，如字典中只有 positive、negative、neutral，則它們的向量可表示為：

$$\text{positive} = [1,0,0]$$

$$\text{negative} = [0,1,0]$$

$$\text{neutral} = [0,0,1]$$

上述兩種詞嵌入的做法中有幾種缺點，當字典中的詞彙隨著增大而不是只有三個詞彙可能為幾千、幾萬個詞彙，這樣就會導致向量維度也會增加，對電腦的負擔也就越大。在獨熱編碼中，假設這個字典中有1000個詞彙，代表一個向量裡有999個0，導致稀疏性的問題。在自然語言處理中語言本身的語意非常重要，由於兩種詞嵌入沒有順序性的關係，所以忽略了單詞本身的語意，這樣會導致判斷的準確率下降。

為了解決這些缺點，後續研究提出了 Word2Vec (Mikolov, T., Chen, K., Corrado, G., & Dean, J. 2013)模型以及 Glove (Pennington, J., Socher, R., & Manning, C. D. 2014)模型。

Word2Vec 由 Google 於2013年 Tomas Mikolov 等人提出，主要目的是將字詞轉換成向量表達它們的語意，透過學習大量文本資料，讓語意相似的單字有較近的距離，Word2Vec 中有兩種模型，分別為 CBOW 模型(Continuous Bag-of-Word Model)(Mikolov, T., Chen, K., Corrado, G., & Dean, J. 2013)與 Skip-gram 模型(Continuous Skip-gram Model) (Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. 2013.)，CBOW 給定上下文後，來預測輸入的字詞，而 Skip-gram 則是給定輸入的字詞後，來預測上下文，如圖3.6所示，如有兩個不同的單詞有相似的上下文，在模型訓練後嵌入向量則會十分相似，在進行自然語言處理工作時，對於準確率來說效果非常好。

Glove (Pennington, J., Socher, R., & Manning, C. D. 2014, October.)是基於全局詞頻統計的詞表徵工具，從一個單詞表達成一個向量，再利用向量捕捉語意的特性，透過向量的計算，來計算出單詞之間的語意的相似性，Glove 結合 LSA 和 Word2Vec 優點，利用語料庫全局統計的訊息，不僅訓練更快，對於大規模或者

小規模的語料庫都能擁有良好的表現，史丹佛 Glove 團隊也會定期的更新詞典和詞向量，在官方網站中已經有提供多種訓練完成的詞向量讓每個使用者使用，出處分別有維基百科、推特、Common Crawl、Gigaword 5，根據詞彙數量分別有400K、1.2M、1.9M、2.2M，本實驗使用 glove.6B<sup>2</sup>資料檔案，6B 代表使用60億的單字語料庫訓練，共含400K的詞彙，另外也包含50、100、200、300不同維度提供使用，非常適合本研究的實驗。

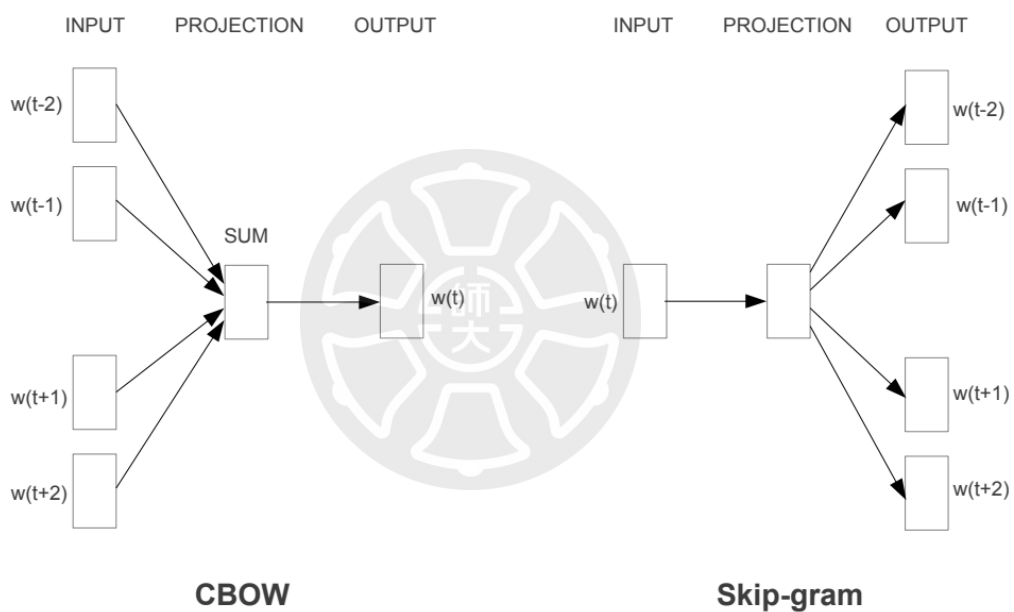


圖 3.6：CBOW 與 Skip-gram 模型架構圖

(圖片參考自：Mikolov, T., Chen, K., Corrado, G., & Dean, J. 2013)

在將資料放進模型訓練之前，需要先將處理好的資料都轉換成向量的形式，也就是將文字轉換成數字型態來表達，電腦才能對這個資料進行處理，否

<sup>2</sup> <https://nlp.stanford.edu/projects/glove/>

則無法輸入模型中，本研究使用 Word2Vec 以及 Glove 去對每一種模型做處理，再分別比較兩種不同的方法的差異性。

由於世界上的詞彙數量極多，系統所訓練的模型不可能涵蓋這麼多，所以如果沒有找到對應的詞彙，則會使用零填充(Zero-padding)來進行處理，當作未知詞的向量，如圖3.7所示。

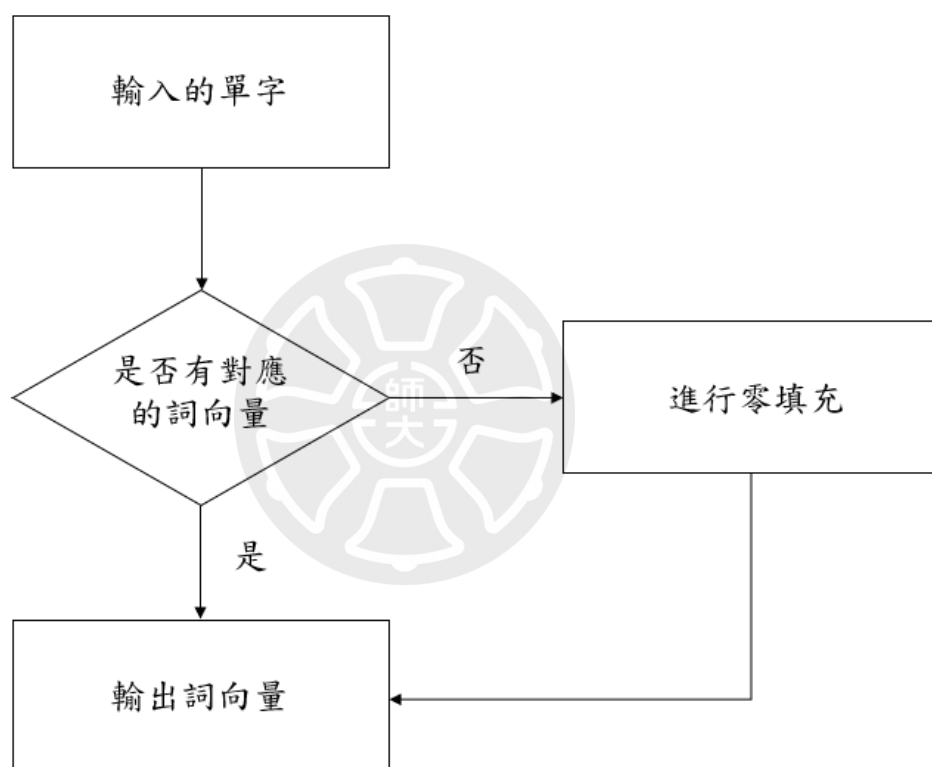


圖 3.7：詞彙轉換詞向量流程圖

## 第八節 雙向長短期記憶

雙向長短期記憶 (Bi-directional Long-Short Term Memory) 簡稱 Bi-LSTM (Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., & Dyer, C. 2016) 由前向 LSTM 與後向 LSTM 結合，LSTM 透過訓練可以學習到需要記憶哪些重要的資訊或者遺忘那些資訊，但這樣的方法造成 LSTM 只能由單向去找出資訊，隨著時間序列成長，越後面的詞會比前面的詞更為重要，無法去捕捉更好的關係，因此透過 Bi-LSTM 來解決，Bi-LSTM 從左到右做一次 LSTM 也從右到左再做一次 LSTM，由上一層的輸出結果給下一層的輸入持續學習，這樣的學習方法更可以捕捉到相關的關係，如圖3.8所示。

而本研究的實驗將會比較雙向長短期記憶模型與長短期記憶模型，實驗結果將於第四章呈現。

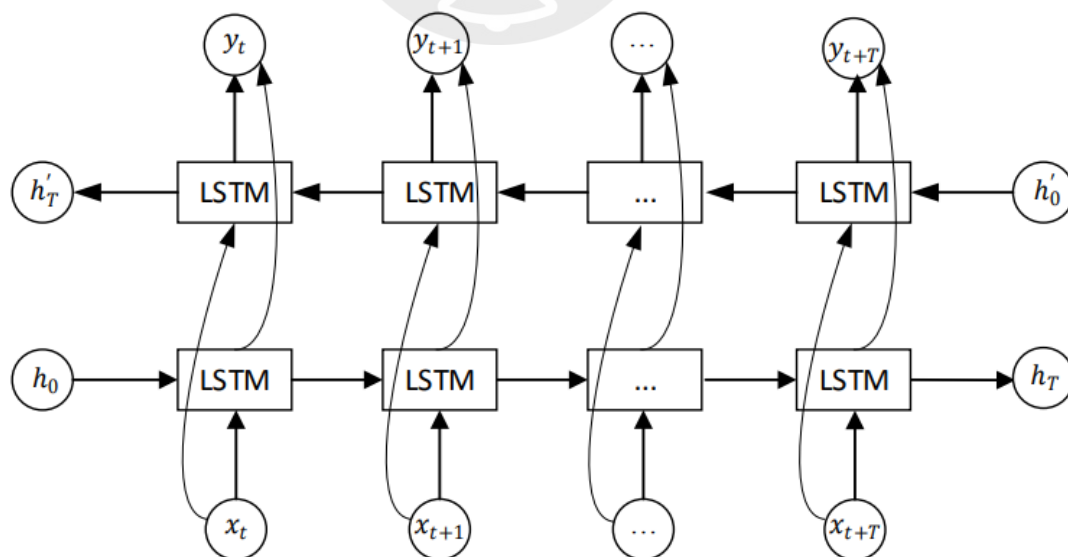


圖 3.8：雙向長短期記憶模型架構

(圖片參考自：Xiang, J., Qiu, Z., Hao, Q., & Cao, H. 2020)

## 第九節 Encoder-Decoder

Encoder-Decoder 並不是一個模型，而是一個框架。Encoder-Decoder 架構如圖3.9所示，在自然語言處理裡，經常應用在機器翻譯、文章摘要等等，將對話或者文本，轉化成數學的問題，再透過不同的方法求解數學問題，解決現實的問題，但是 Encoder-Decoder 最大的問題就是訊息的缺失，也就是說當輸入的訊息過長，會導致訊息被稀釋掉。

為了解決 Encoder-Decoder 訊息缺失的問題，會再加上 Attention 機制來解決這個問題，經常與 Encoder-Decoder 架構一起使用，產生一個注意力的範圍，來代表接下來的輸出要關注輸入中那些部分，根據關注的區域產生下一個輸出。加上 Attention 的差別在於不用在編碼器中將所有的輸入都放進一個固定的長度向量中，這樣的優點在產生每一個輸出，都能充分利用輸入的訊息。

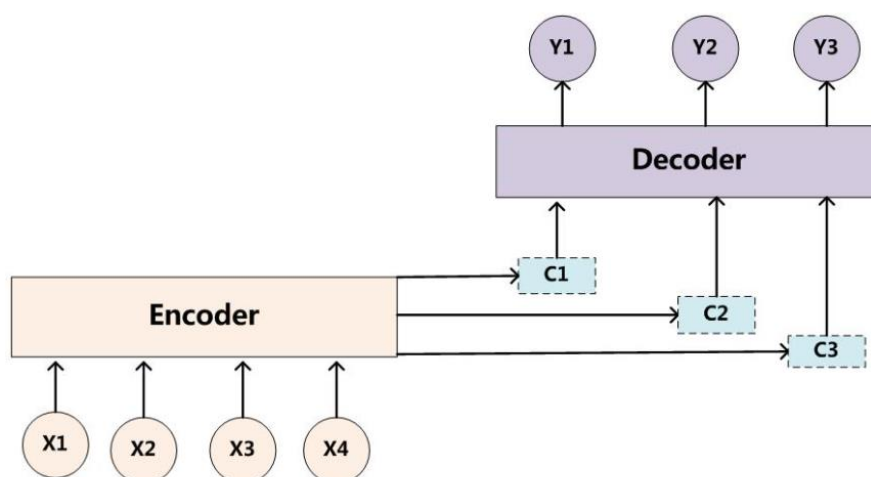


圖 3.9：Encoder-Decoder 架構

(圖片參考自：<https://blog.csdn.net/u014595019/article/details/52826423>)

在本實驗的過程中，採用一層的 Encoder-Decoder 的準確度並沒有達到所預期的分數，且在分類面向詞極性時常常判斷錯誤，故本研究嘗試使用兩層的 Encoder-Decoder，並且與一層 Encoder-Decoder 比較差異性，一層的 Encoder-Decoder 模型架構如圖3.10所示。

實驗結果如表3.10到3.12所示，在訓練模型時與一層的 Encoder-Decoder 相比兩層的 Encoder-Decoder 對於 Training loss 在每10秒時都有明顯的在持續穩定下降，如表3.10所示，而在準確度的部份，在餐廳評論或者筆記型電腦評論中，兩層的 Encoder-Decoder 與一層的 Encoder-Decoder 相比也提高許多，如表3.11、表3.12所示。

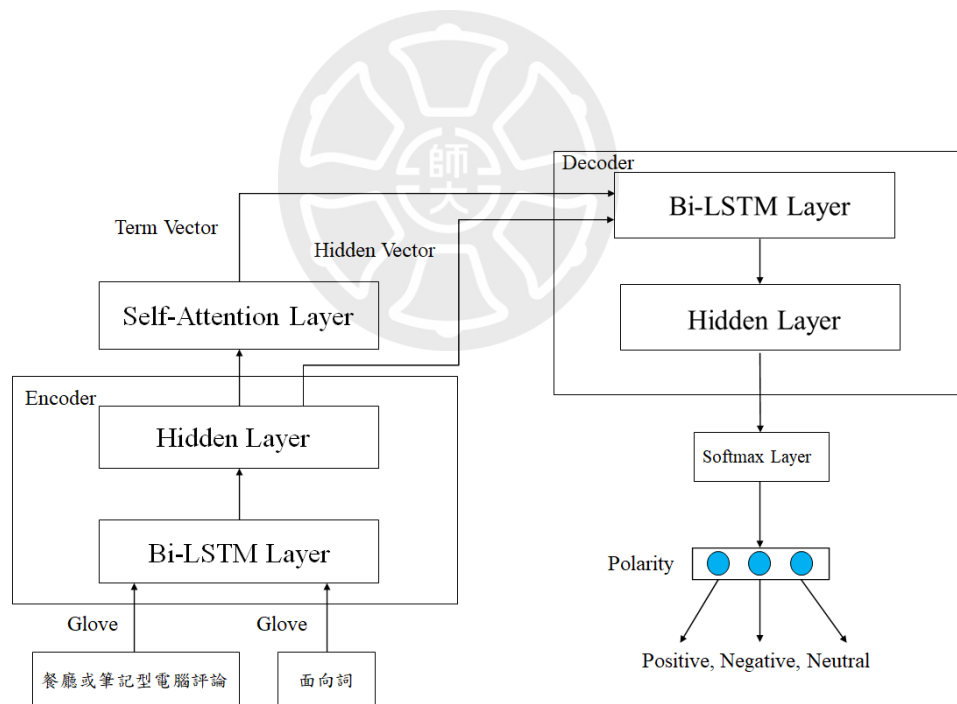


圖 3.10：One-level Encoder-Decoder 模型架構

表 3.10：Encoder-Decoder 與 Two-level Encoder-Decoder 之 Training Loss 比較

Time (seconds)	Loss	
	One-level Encoder-Decoder	Two-level Encoder-Decoder
10	0.4627	0.4216
20	0.4295	0.3792
30	0.3887	0.3610
40	0.3716	0.3545
50	0.3684	0.3392

表 3.11：Encoder-Decoder 與 Two-level Encoder-Decoder 餐廳評論準確度比較

Accuracy	
Encoder-Decoder	Two-level Encoder-Decoder
0.7817	0.8267

表 3.12：Encoder-Decoder 與 Two-level Encoder-Decoder  
筆記型電腦評論準確度比較

Accuracy	
Encoder-Decoder	Two-level Encoder-Decoder
0.7581	0.7863

## 第四章 實驗結果

### 第一節 開發環境

本研究以 python 語言撰寫程式，實驗開發環境如表4.1及表4.2。

表 4.1：硬體開發環境

項目	規格
中央處理器	Intel(R) Core(TM) i9-9900K CPU @ 3.60GHZ
記憶體	16GB
顯示卡	NVIDIA GeForce RTX 2080 Ti

表 4.2：軟體開發環境

項目	規格
作業系統	Windows 10
程式語言	Python
框架	Tensorflow 1.12.0
開發環境	Anaconda、Jupyter

## 第二節 實驗結果

本研究嘗試多種方法，測試餐廳以及筆記型電腦評價，從一開始最基本的長短期記憶模型，到雙向長短期記憶模型，以及將雙向長短期記憶模型中加上自注意力機制以及條件隨機場，最後使用 Two-Level Encoding，逐一用不同方法測試餐廳及筆記型電腦中的準確度。故本研究實驗模型測試包含五種不同的模型，其架構圖如圖4.1到4.5所示。

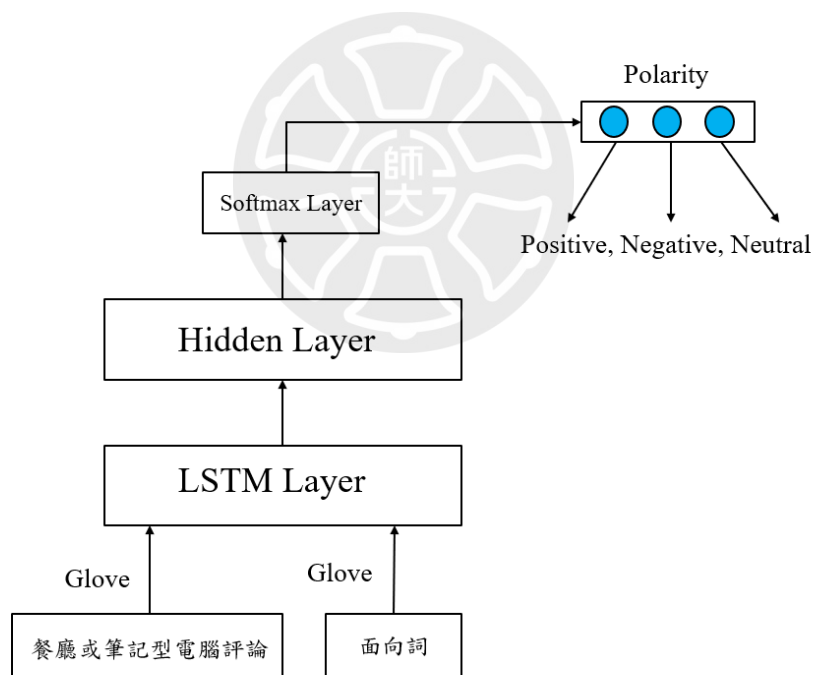


圖 4.1：LSTM 模型架構

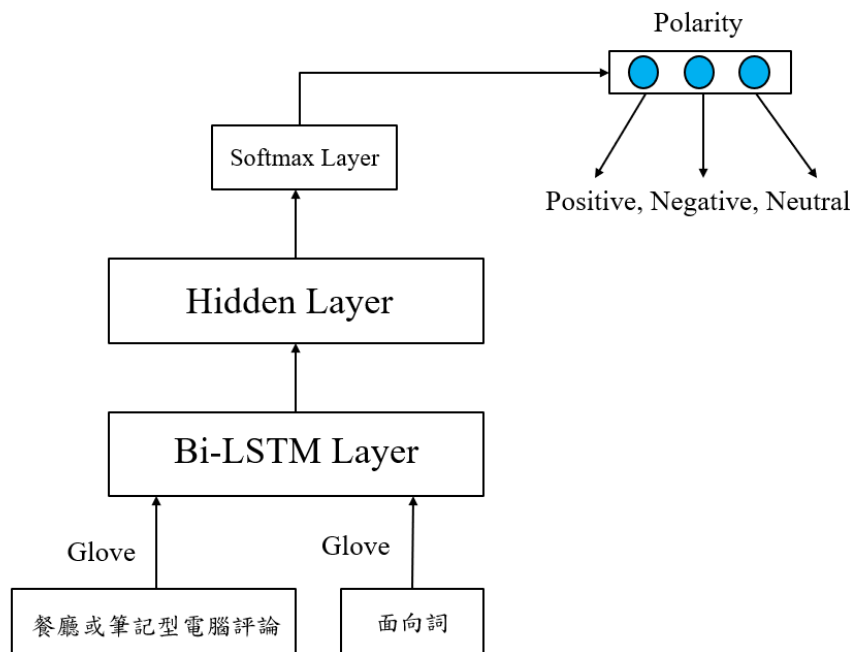


圖 4.2 : Bi-LSTM 模型架構

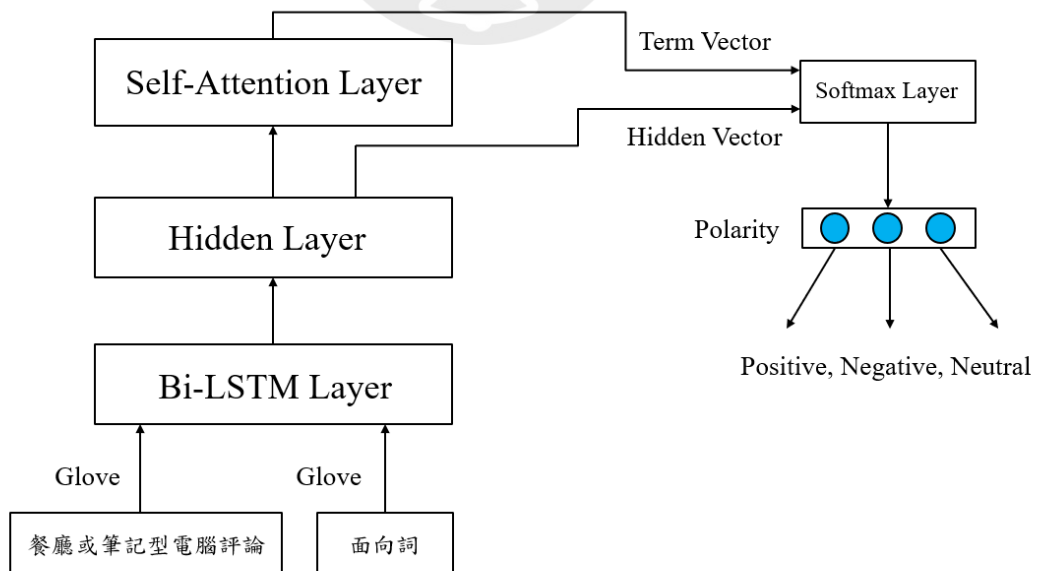


圖 4.3 : Bi-LSTM + Self-Attention 模型架構

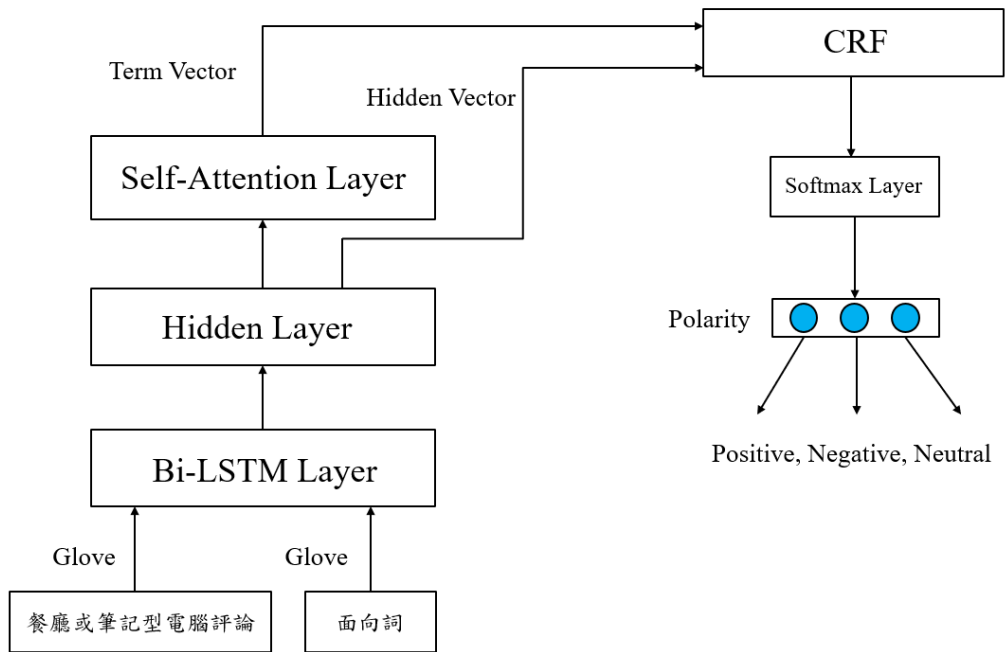


圖 4.4 : Bi-LSTM + Self-Attention + CRF 模型架構

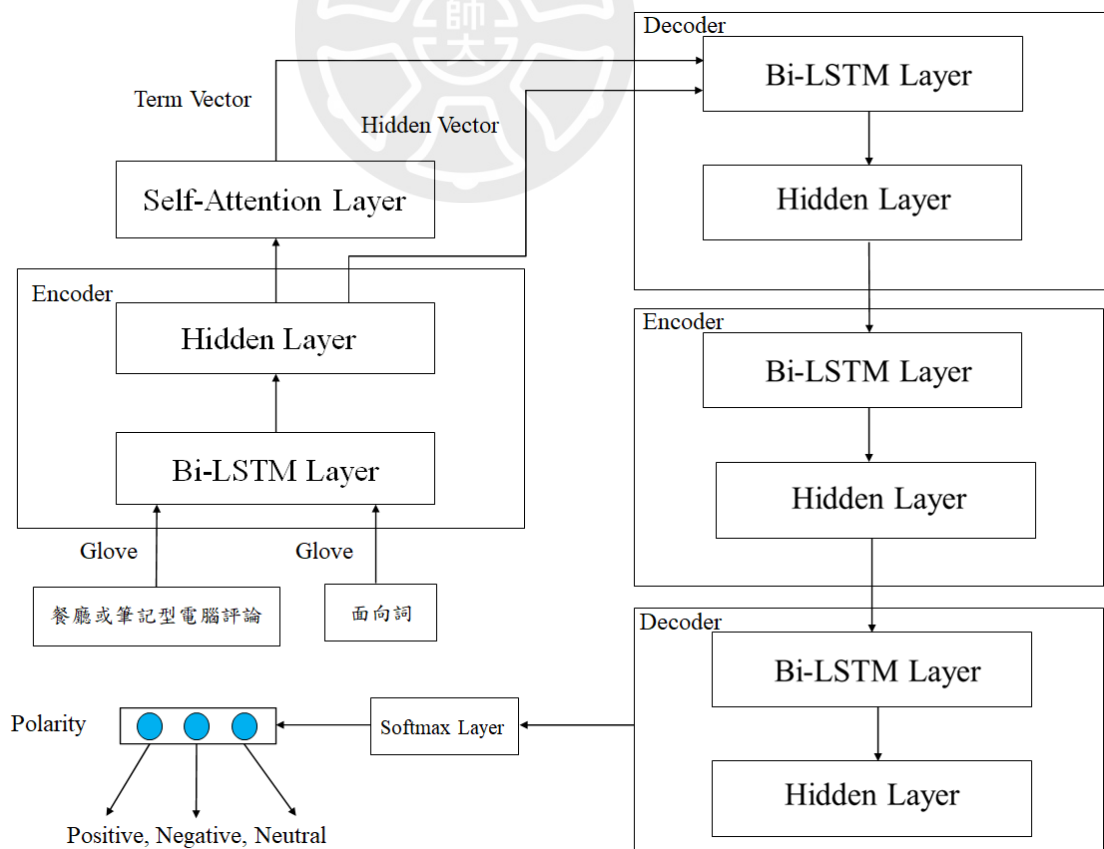


圖 4.5 : Two-Level Encoding 模型架構

在轉換成詞向量的部分運用兩種不同的詞嵌入方法來測試每種模型不同的準確度，分別為 Word2Vec 以及 Glove，以及對於不同的維度進行實驗，以便找出在多少維度下模型的準確率為最佳，由表4.3到表4.6中，可以看出每一個模型在維度200到300之間準確度為最佳，同時本實驗也測試每一個模型在不同的維度下所訓練的時間，如表4.7到表4.10所示。

表 4.3：使用 Word2vec 詞嵌入前處理五種不同模型準確度(餐廳)

模型 \ 資料集	維度			
	50	100	200	300
LSTM	0.6955	0.7062	0.7235	0.7385
Bi-LSTM	0.7184	0.7324	0.7410	0.7517
Bi-LSTM + Self-Attention	0.7458	0.7537	0.7602	0.7643
Bi-LSTM + Self-Attention + CRF	0.7687	<b><u>0.7759</u></b>	0.7781	0.7808
Two-Level Encoding	<b><u>0.7764</u></b>	0.7753	<b><u>0.8056</u></b>	<b><u>0.8198</u></b>

表 4.4：使用 Word2vec 詞嵌入前處理五種不同模型準確度(筆記型電腦)

模型 \ 資料集	維度			
	50	100	200	300
LSTM	0.6552	0.6694	0.6957	0.6852
Bi-LSTM	0.6744	0.6853	0.6958	0.7121
Bi-LSTM + Self-Attention	0.6982	0.7181	0.7389	0.7260

Bi-LSTM + Self-Attention + CRF	0.7427	0.7407	0.7621	0.7508
Two-Level Encoding	<b><u>0.7532</u></b>	<b><u>0.7597</u></b>	<b><u>0.7639</u></b>	<b><u>0.7837</u></b>

表 4.5：使用 Glove 詞嵌入前處理五種不同模型準確度(餐廳)

模型 \ 資料集	維度			
	50	100	200	300
LSTM	0.7217	0.7241	0.7443	0.7452
Bi-LSTM	0.7357	0.7468	0.7501	0.7524
Bi-LSTM + Self-Attention	0.7321	0.7458	0.7552	0.7679
Bi-LSTM + Self-Attention + CRF	<b><u>0.7611</u></b>	<b><u>0.7814</u></b>	0.7743	0.7852
Two-Level Encoding	0.7550	0.7803	<b><u>0.8267</u></b>	<b><u>0.8182</u></b>

表 4.6：使用 Glove 詞嵌入前處理五種不同模型準確度(筆記型電腦)

模型 \ 資料集	維度			
	50	100	200	300
LSTM	0.6414	0.6683	0.6891	0.6992
Bi-LSTM	0.6777	0.6950	0.7113	0.7167
Bi-LSTM + Self-Attention	0.6843	0.7092	0.7266	0.7411
Bi-LSTM + Self-Attention + CRF	0.7302	0.7422	0.7569	0.7647
Two-Level Encoding	<b><u>0.7433</u></b>	<b><u>0.7681</u></b>	<b><u>0.7727</u></b>	<b><u>0.7863</u></b>

表4.7：使用 Word2vec 詞嵌入前處理五種不同模型訓練時間(餐廳)

模型 \ 訓練時間	維度			
	50	100	200	300
LSTM	41秒	43秒	44秒	46秒
Bi-LSTM	44秒	47秒	48秒	50秒
Bi-LSTM + Self-Attention	1分11秒	1分14秒	1分18秒	1分23秒
Bi-LSTM + Self-Attention + CRF	1分34秒	1分38秒	1分44秒	1分51秒
Two-Level Encoding	4分47秒	4分54秒	5分11秒	5分28秒

表4.8 使用 Word2vec 詞嵌入前處理五種不同模型訓練時間(筆記型電腦)

模型 \ 訓練時間	維度			
	50	100	200	300
LSTM	34秒	35秒	37秒	37秒
Bi-LSTM	35秒	37秒	39秒	41秒
Bi-LSTM + Self-Attention	50秒	54秒	59秒	1分08秒
Bi-LSTM + Self-Attention + CRF	1分04秒	1分07秒	1分16秒	1分24秒
Two-Level Encoding	4分23秒	4分35秒	4分46秒	4分58秒

表4.9：使用 Glove 詞嵌入前處理五種不同模型訓練時間(餐廳)

模型 \ 訓練時間	維度			
	50	100	200	300
LSTM	53秒	54秒	57秒	59秒
Bi-LSTM	56秒	1分00秒	1分06秒	1分13秒
Bi-LSTM + Self-Attention	1分32秒	1分39秒	1分47秒	1分56秒
Bi-LSTM + Self-Attention + CRF	1分50秒	1分57秒	2分06秒	2分15秒
Two-Level Encoding	5分23秒	5分29秒	5分45秒	5分58秒

表4.10：使用 Glove 詞嵌入前處理五種不同模型訓練時間(筆記型電腦)

模型 \ 訓練時間	維度			
	50	100	200	300
LSTM	44秒	47秒	51秒	56秒
Bi-LSTM	52秒	57秒	1分02秒	1分10秒
Bi-LSTM + Self-Attention	1分21秒	1分24秒	1分35秒	1分42秒
Bi-LSTM + Self-Attention + CRF	1分38秒	1分44秒	1分53秒	2分04秒
Two-Level Encoding	5分11秒	5分17秒	5分29秒	5分40秒

### 第三節 分析與討論

從表4.3到表4.6中實驗結果顯示，在詞嵌入的部分，使用 Glove 詞嵌入，與 Word2Vec 詞嵌入比較，準確度都較為良好；在模型的部分，由於長短期記憶模型只能單向的去尋找資訊，而在雙向長短期記憶模型中可以從右到左也從左到右去尋找資訊，實驗結果也表明效果比長短期記憶模型模型準確度來得高，也可以從實驗結果發現，雙向長短期記憶模型結合注意力機制在分析面向詞情感取得顯著的提升，由此可得知雙向長短期記憶模型可藉由加入面向詞資訊提升準確度。而本實驗也對不同的維度進行比較，在維度50時準確度效果為最差，而在200到300之間的維度，準確度較為良好，在表4.5可以看出維度在200時，準確度比維度在300時來的更佳，若是我們再繼續提升維度可能會造成準確度降低。

本實驗也與 SemEval-2014主辦單位所舉辦的比賽，去與其他當年所參加的團隊做準確度的比較，在 SemEval-2014當年比賽中，第一名 DCU (Wagner, J., Arora, P., Cortes, S., Barman, U., Bogdanova, D., Foster, J., & Tounsi, L. 2014)團隊與第二名 NRC-Can (Kiritchenko, S., Zhu, X., Cherry, C., & Mohammad, S. 2014)團隊，都是使用支援向量機(Support Vector Machine)(Cortes, C., & Vapnik, V. 1995)，簡稱 SVM，可以從準確度看出來，在當年比賽中使用 SVM 方法不管是對於餐廳或者是筆記型電腦評論，都得到一個非常好的準確度，但 SVM 方法需提取多種特徵進行訓練，與類神經網路方法訓練方式非常不同，在本研究中採取了雙向長短期記憶模型加上自注意力機制並結合了 Two-level Encoding，希望捕捉到更好的文本上下關係，進而判別每個句子中面向詞代表哪一種極性分類。

從表4.7及4.8中可以看出本論文提出的實驗方法得到良好的實驗結果。

表 4.11：SemEval-2014 餐廳評論前十名團隊與本實驗之準確度

名次	團隊	Accuracy
1	DCU	0.8095
2	NRC-Can	0.8015
3	UWB	0.7768
4	XRCE	0.7768
5	SZTE-NLP	0.7522
6	UNITOR	0.7495
7	UBham	0.7460
8	USF	0.7319
9	UNITOR	0.7248
10	SeemGO	0.7231
	Our	<b>0.8267</b>

表 4.12：SemEval-2014 筆記型電腦評論前十名團隊與本實驗之準確度

名次	團隊	Accuracy
1	DCU	0.7048
2	NRC-Can	0.7048
3	SZTE-NLP	0.6697
4	UBham	0.6666
5	UWB	0.6666
6	Isis_lif	0.6452
7	USF	0.6452

8	SNAP	0.6406
9	UNITOR	0.6299
10	UWB	0.6253
	Our	<b>0.7863</b>



## 第五章 結論與未來展望

### 第一節 結論

情緒分析是非常值得探討的一項議題，對於表達情感方面是非常重要的，在現今社會上許多人會將服務評價、產品的優劣度評價當作一項購買指標或者觀顧指標，讓使用者能夠在過去的評價中找到自己所想尋找到的資訊，而在本研究中我們也探討對於餐廳評論以及筆記型電腦評論的每個句子中不同的面向詞，每一個面向詞代表著什麼樣的評價。

在實驗的過程中，並不是所有面向詞都判斷的十分順利，在 SemEval-2014 官方資料集當中，常常會有一些面向詞難以分析它的情感極性造成情感極性判斷失敗，而導致準確度下降。所以本實驗測試許多模型，不同的詞嵌入方法，以及做了不同的前處理方法，嘗試讓面向詞的情感極性判斷更為準確，而我們也發現在 SemEval-2014 官方資料集當中，不管是餐廳評論或者是筆記型電腦評論，訓練集資料筆數並不是非常多，尤其在筆記型電腦評論中更是明顯，導致在訓練的過程中，無法更有效的訓練每一個面相詞對於這句話的含意，準確度也無法更有效的提升，若能夠增加訓練資料集的總數，對於訓練的成果想必會大幅提升。

## 第二節 未來研究方向

在實驗的過程中，訓練面向詞是非常重要的，由於目前可搜尋到的公開提供開放的相關資料集只有 SemEval 和 Sentihood，且資料筆數並不是很大量，未來希望能夠與各大餐廳互相的去對於評論做面向詞的分析，使面向詞擁有更多層面的意義，並增加訓練資料的總筆數，在處理資料時也會更加容易，在判斷準確度的部份也會有著良好的表現，也希望能夠運用更多的模型，例如：TD-LSTM、TC-LSTM、ATAE-LSTM 等等，驗證不同的訓練模型在準確度部分是否能夠達到不一樣的效果，以及再加上各式不同的前處理方法於不同的模型，期望能夠突破目前的準確度。



## 參考文獻

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3), 273-297.

Gao, Z., Feng, A., Song, X., & Wu, X. (2019). Target-dependent sentiment classification with BERT. *Ieee Access*, 7, 154290-154299.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8), 1735-1780.

Kiritchenko, S., Zhu, X., Cherry, C., & Mohammad, S. (2014, August). Nrc-canada-2014: Detecting aspects and sentiment in customer reviews. In *Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014)* (pp. 437-442).

Lafferty, J., McCallum, A., & Pereira, F. C. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., & Dyer, C. (2016). Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26.

Nguyen, H. T., & Le Nguyen, M. (2018, November). Effective attention networks for aspect-level sentiment classification. In *2018 10th International Conference on Knowledge and Systems Engineering (KSE)* (pp. 25-30). IEEE.

Pennington, J., Socher, R., & Manning, C. D. (2014, October). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 1532-1543).

Pontiki, M., Galanis, D., Pavlopoulos, J., Papageorgiou, H., Androutsopoulos, I., & Manandhar, S. (2014). SemEval-2014 Task 4: Aspect based sentiment analysis. In *Proceedings of the International Workshop on Semantic Evaluation, SemEval '14*, 27–35.

Tang, D., Qin, B., Feng, X., & Liu, T. (2015). Effective LSTMs for target-dependent sentiment classification. *arXiv preprint arXiv:1512.01100*.

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.

Varsamopoulos, S., Bertels, K., & Almudever, C. G. (2018). Designing neural network based decoders for surface codes. *arXiv preprint arXiv:1811.12456*.

Wagner, J., Arora, P., Cortes, S., Barman, U., Bogdanova, D., Foster, J., & Tounsi, L. (2014). Dcu: Aspect-based polarity classification for semeval task 4.

Wallach, H. M. (2004). Conditional random fields: An introduction. *Technical Reports (CIS)*, 22.

Wagner, W. (2010). Steven bird, ewan klein and edward loper: Natural language processing with python, analyzing text with the natural language toolkit. *Language Resources and Evaluation*, 44(4), 421-424.

Xie, J., Chen, B., Gu, X., Liang, F., & Xu, X. (2019). Self-attention-based BiLSTM model for short text fine-grained sentiment classification. *IEEE Access*, 7, 180558-180570.

Xiang, J., Qiu, Z., Hao, Q., & Cao, H. (2020). Multi-time scale wind speed prediction based on WT-bi-LSTM. In *MATEC Web of Conferences* (Vol. 309, p. 05011). EDP Sciences.

Zaremba, W., Sutskever, I., & Vinyals, O. (2014). Recurrent neural network regularization. *arXiv preprint arXiv:1409.2329*.

Zhang, Y., Jin, R., & Zhou, Z. H. (2010). Understanding bag-of-words model: a statistical framework. *International Journal of Machine Learning and Cybernetics*, 1(1), 43-52.

