

國立中央大學

資訊工程學系
碩士論文

以同反義詞典調整的詞向量對下游自然語言任務
影響之實證研究

Adjusting Word Embeddings Based on the Dictionary of
Synonyms and Antonyms and Its Influence on
Downstream NLP Tasks – an Empirical Study

研究生：陳堃澤

指導教授：陳弘軒 博士

中華民國一百一十年六月

國立中央大學圖書館學位論文授權書

填單日期：110 / 08 / 04

2019.9 版

授權人姓名	陳懿澤	學號	108522042
系所名稱	資訊工程學系碩士班	學位類別	<input checked="" type="checkbox"/> 碩士 <input type="checkbox"/> 博士
論文名稱	以同反義詞典調整的詞向量對下游自然語言任務影響之實證研究	指導教授	陳弘軒

學位論文網路公開授權

授權本人撰寫之學位論文全文電子檔：

- 在「國立中央大學圖書館博碩士論文系統」
 - (☒)同意立即網路公開
 - ()同意 於西元_____年_____月_____日網路公開
 - ()不同意網路公開，原因是：_____
- 在國家圖書館「臺灣博碩士論文知識加值系統」
 - (☒)同意立即網路公開
 - ()同意 於西元_____年_____月_____日網路公開
 - ()不同意網路公開，原因是：_____

依著作權法規定，非專屬、無償授權國立中央大學、台灣聯合大學系統與國家圖書館，不限地域、時間與次數，以文件、錄影帶、錄音帶、光碟、微縮、數位化或其他方式將上列授權標的基於非營利目的進行重製。

學位論文紙本延後公開申請 (紙本學位論文立即公開者此欄免填)

本人撰寫之學位論文紙本因以下原因將延後公開

- 延後原因
 - ()已申請專利並檢附證明，專利申請案號：_____
 - ()準備以上列論文投稿期刊
 - ()涉國家機密
 - ()依法不得提供，請說明：_____

• 公開日期：西元_____年_____月_____日

※繳交教務處註冊組之紙本論文(送繳國家圖書館)若不立即公開，請加填「國家圖書館學位論文延後公開申請書」

研究生簽名：陳懿澤

指導教授簽名：陳弘軒

國立中央大學碩士班研究生
論文指導教授推薦書

資訊工程學系碩士班 學系/研究所 陳堃澤 研究生

所提之論文 以同反義詞典調整的詞向量對下游自然語言任務影響之實證研究

係由本人指導撰述，同意提付審查。

指導教授 陳弘毅 (簽章)

2021 年 7 月 8 日

國立中央大學碩士班研究生
論文口試委員審定書

資訊工程學系碩士班 學系/研究所 陳堃澤 研究生

所提之論文 以同反義詞典調整的詞向量對下游自然語言任務影響之實證研究

經由委員會審議，認定符合碩士資格標準。

學位考試委員會召集人

黃瀚章

委

員

張嘉惠

洪弘軒

110 年 7 月 27 日

以同反義詞典調整的詞向量對下游自然語言任務 影響之實證研究

摘要

「向量」的概念目前已被廣泛運用在機器學習的領域裡。在自然語言處理領域中，學者將輸入的字詞轉換為「詞向量」，以讓電腦進行方便且有效的模型訓練。而現今學者也致力於研究如何讓詞向量更能表達出符合文本與詞庫中的字詞關係，大致上能將這些訓練方法分為兩類，一是使用文本加上詞庫中的字詞關係同時訓練，另一種則是對現今已存在的詞向量加上字詞關係同時訓練再進行訓練。

本研究的訓練方法屬於第二類，我們透過自注意力機制對現有的詞向量進行調整，使詞向量能學習到詞庫中的同義詞反義詞關係。實驗發現透過自注意力機制訓練出的新的詞向量更符合詞庫中的字詞關係的詞向量，但將此詞向量對下游的自然語言處理任務進行處理時，卻得到比調整前的詞向量更差的結果。

關鍵字：調整詞向量, 同義詞, 反義詞, 自然語言處理任務

Adjusting Word Embeddings Based on the Dictionary of Synonyms and Antonyms and Its Influence on Downstream NLP Tasks – an Empirical Study

Abstract

The concept of "vector" has been widely used in machine learning. For example, in the field of natural language processing, researchers convert words into vectors, also known as word embeddings, so that computers can access a fixed-length vector as features for model training. Researchers also study methodologies to generate word embeddings that better express the semantic relationship between the words specified in the lexicon. These methods can be divided into two categories. The first type is to generate word embeddings by simultaneously considering both the word co-appearance relationship in a given corpus and lexicon knowledge, e.g., synonyms or antonyms. The second type is to adjust existing (pre-trained) word embeddings with lexicon knowledge.

We study the second type of method in this thesis. We adjust the pre-trained word embeddings through a self-attention mechanism so that the word embeddings can preserve the relationship between synonyms and antonyms in the lexicon. Experimental results show that the adjusted word embeddings indeed better keep synonym and antonym information. However, if these word embeddings are used as the input of the downstream

natural language processing task, it will get worse results than the word embeddings before adjustment.

Keywords: Adjusting Word Embedding, Synonyms, Antonyms, Natural Language Processing Task

目錄

	頁次
摘要	ix
Abstract	xi
目錄	xiii
一、緒論	1
二、相關研究	5
2.1 使用文本及詞庫中的字詞關係同時訓練詞向量	5
2.2 使用詞庫中的字詞關係調整預訓練詞向量	6
2.2.1 Retrofitting Word Vectors to Semantic Lexicons	6
2.2.2 ATTRACT-REPEL.....	7
2.2.3 自注意力機制	8
三、研究模型及方法	11
3.1 Task 1: 調整詞向量.....	11
3.1.1 資料前處理	11
3.1.2 自注意力機制 (Self-Attention Mechanism)	12
3.1.3 損失函數	13
3.2 Task 2: 自然語言處理任務.....	14
3.2.1 長短期記憶模型 (LSTM)	15

四、	實驗結果與分析	17
4.1	Task 1: 調整詞向量.....	17
4.1.1	模型設置	17
4.1.2	實驗資料集	18
4.1.3	實驗結果	20
4.2	Task 2: 自然語言處理任務.....	35
4.2.1	模型設置	35
4.2.2	IMDb.....	35
4.2.3	US Airline.....	42
五、	總結	47
5.1	結論	47
5.2	未來展望	48
	參考文獻	49
	附錄 A 實驗程式碼	53

圖目錄

	頁次
3.1 無方向性模型架構	11
3.2 自注意力機制計算流程	12
4.1 Top-100 similar words of “his” in GloVe _{42B}	33
4.2 Top-100 similar words of “his” in our work.	34
4.3 詞向量 GloVe _{6B} 及以 GloVe _{6B} 調整的詞向量 IMDb 預測之 結果。此圖調整後之詞向量未經 finetune。	37

表目錄

	頁次
4.1 使用 FrameNet _{syn} 資料集調整預訓練詞向量 GloVe。數據標示部分，粗體為最佳數據。	21
4.2 使用 PPDB _{syn} 資料集調整預訓練詞向量 GloVe。數據標示部分，粗體為最佳數據。	21
4.3 使用 PPDB _{syn+ant} 資料集調整預訓練詞向量 GloVe。數據標示部分，粗體為最佳數據。	22
4.4 使用 WordNet _{syn} 資料集調整預訓練詞向量 GloVe。數據標示部分，粗體為最佳數據。	22
4.5 使用 WordNet _{syn+ant} 資料集調整預訓練詞向量 GloVe。數據標示部分，粗體為最佳數據。	23
4.6 使用不同同反義詞資料集調整預訓練詞向量 Word2Vec。數據標示部分，粗體為最佳數據。	24
4.7 使用不同同反義詞資料集調整預訓練詞向量 fastText。數據標示部分，粗體為最佳數據。	25
4.8 使用不同 PPDB 資料集調整預訓練詞向量 GloVe _{6B} 。數據標示部分，粗體為最佳數據。	26
4.9 使用不同 PPDB 資料集調整預訓練詞向量 GloVe _{42B} 。數據標示部分，粗體為最佳數據。	27
4.10 使用不同 PPDB 資料集調整預訓練詞向量 Word2Vec。數據標示部分，粗體為最佳數據。	27

4.11 使用不同 PPDB 資料集調整預訓練詞向量 fastText。數據標示部分，粗體為最佳數據。	28
4.12 使用不同 Lexicon 資料集調整預訓練詞向量 GloVe _{6B} 。數據標示部分，粗體為最佳數據。	28
4.13 使用不同 Lexicon 資料集調整預訓練詞向量 GloVe _{42B} 。數據標示部分，粗體為最佳數據。	29
4.14 使用不同 Lexicon 資料集調整預訓練詞向量 Word2Vec。數據標示部分，粗體為最佳數據。	29
4.15 使用不同 Lexicon 資料集調整預訓練詞向量 fastText。數據標示部分，粗體為最佳數據。	30
4.16 調整前後的詞向量對相似度任務 (SimLex-999) 的實驗結果，表格中數據為斯皮爾曼等級相關係數。數據標示部分，粗體為最佳數據。	31
4.17 調整結果展示	32
4.18 詞向量 GloVe _{6B} 及以 GloVe _{6B} 調整的詞向量 IMDb 預測之結果。數據標示部分，粗體為最佳數據。	38
4.19 詞向量 GloVe _{42B} 及以 GloVe _{42B} 調整的詞向量 IMDb 預測之結果。數據標示部分，粗體為最佳數據。	39
4.20 詞向量 Word2Vec 及以 Word2Vec 調整的詞向量 IMDb 預測之結果。數據標示部分，粗體為最佳數據。	40
4.21 詞向量 fastText 及以 fastText 調整的詞向量 IMDb 預測之結果。數據標示部分，粗體為最佳數據。	41
4.22 詞向量 GloVe _{6B} 及以 GloVe _{6B} 調整的詞向量對 US airline 預測之結果。數據標示部分，粗體為最佳數據。	43
4.23 詞向量 GloVe _{42B} 及以 GloVe _{42B} 調整的詞向量對 US airline 預測之結果。數據標示部分，粗體為最佳數據。	44
4.24 詞向量 Word2Vec 及以 Word2Vec 調整的詞向量對 US airline 預測之結果。數據標示部分，粗體為最佳數據。	45

4.25 詞向量 fastText 及以 fastText 調整的詞向量對 US airline 預測之結果。數據標示部分，粗體為最佳數據，方框為表 現較調整前的詞向量好的數據。	46
-----------------------------------------------------------------------------------------------------------	----

一、緒論

在機器學習的領域中，「向量」是一種常用來表示不同的特徵的方式，例如使用者、商品、字詞等等，借此提供電腦進行運算，使模型能夠對其進行訓練。而機器學習的相關學科—自然語言處理，其目的是讓電腦把輸入的語言變成有意思的符號和關係，然後根據目的再處理，此領域也大量使用「向量」的概念對輸入的字詞進行處理，即本篇研究所要探討的詞向量 (word embedding)。目前詞向量已被廣泛運用在自然語言處理的任務中。不過如何將字詞轉換為詞向量，又如何才能有效地轉換並使其能夠有效地學習字詞間的關係，是目前學者積極研究的方向。

把字詞轉換為向量的方法包括 One-hot Encoding，及透過分布假說 (Distributional Hypothesis)[1] 訓練出的各種詞向量表達方式。前者由於轉換成向量之後的維度可能過大、也無法透過轉換之後的向量了解字與字之間的關係，因而延伸出了後者的方法。分布假說根據不同的訓練方法又可分為 Frequency-based 及 Predicted-based 兩類。前者是基於文字在文本中出現的頻率所產生的詞向量，史丹佛大學提出的 GloVe[2] 產生的詞向量即為此類；後者則是透過文字在文本中的上下文來訓練此文字的詞向量，著名代表則是 Google 團隊提出的 Word2Vec[3]。訓練出這些基礎的詞向量之後，為了能夠使訓練出的詞向量更符合自然語言的字詞關係，以提升詞向量在自然語言處理的任務中的表現，學者提出各種方法來訓練出新的詞向量。其中一種是使用詞庫中獲得的字詞關係來加入模型進行訓練。

而本篇論文的研究目標在於，加入詞庫中獲得的字詞關係調整的詞

向量在下游的自然語言處理任務是否有取得較好的表現。根據之前學者們的研究，以此方法調整的詞向量均能學習到更符合自然語言的字詞關係，但只有少數有進行下游的自然語言處理任務的驗證。因此，本研究旨在驗證調整後的詞向量在下游自然語言處理任務究竟有無較好的表現？我們根據 [4] 所提出的模型進行實驗。我們以同義詞與反義詞關係來調整詞向量，並使用調整前後的詞向量作為下游任務的輸入。本研究所選擇的下游自然語言處理任務為情感分析類的任務，其目標是判斷一段敘述所要表達的是正面的還是反面的。我們認為加入同反義詞訓練之後，詞向量應更能判斷出這類的敘述與其正確的標籤 (label)。但經過實驗，我們發現雖然可以使詞向量有效地學習到更多字詞關係的資訊，但其在下游任務的表現卻沒有較調整前的詞向量來得好。

本研究的具體貢獻總結如下：

- 我們借鏡 [4] 所提出的方法，並以新提出的損失函數學習同義詞反義詞等附加資訊以調整詞向量。
- 我們透過測試資料集驗證，調整後的詞向量確實能獲得更良好的字詞關係的資訊。
- 我們進行下游的自然語言處理任務後發現即使詞向量學習到了同義詞反義詞等知識，其在下游的表現不一定更好。

本篇論文共分為六個章節，其架構如下：第一章為緒論，其包含研究動機、研究目的、及本篇論文的貢獻。第二章為與本篇論文有關的其他相關研究。第三章則介紹我們用來訓練詞向量所提出的模型。我們在第四章介紹我們所使用到的資料集。第五章則是我們的實驗及其結果。最

後，第六章為本篇論文的結論及未來展望。

一、緒論

二、 相關研究

如何才能有效地將文字轉換成詞向量並使其能夠學習到更符合自然語言的字詞間的關係，以提升詞向量在自然語言處理任務中的表現，學者提出使用詞庫中獲得的字詞關係來讓提升詞向量的品質。其中，這種訓練方法大致可分為兩類：一為在使用文本訓練詞向量的同時，就加入詞庫中的字詞關係一同訓練 [5]–[13]，使訓練出來的詞向量不僅僅是文本中的上下文關係，也學習到了人類已知的知識；而另一種則是對預訓練的詞向量再進行訓練 [4], [14]–[19]，這種方法訓練時不需要再使用大量文本，而是使用其他學者已經訓練出的詞項量，例如 Glove、Word2Vec 等，再加上字詞關係進行訓練。

2.1 使用文本及詞庫中的字詞關係同時訓練詞向量

此類方法是以文本來訓練詞向量，並在訓練過程中加入訓練過程中加入詞庫中的字詞關係進行更深入的訓練，使詞向量不只是學習文本中的資訊。其中，有一些研究是對語意關係 (semantic constraints) 進行更深入的探討 [5], [10]，其在訓練詞向量的過程中更加強調詞與詞之間的相似性。此外，[13] 將結合典型相關分析 (Canonical Correlation Analysis) 與詞庫中所取得的先備知識訓練出新的詞向量。除了這類利用語意關係進行的研究，[11] 利用對稱圖形 (symmetric patterns) 的概念將反義詞從詞向量上拉遠。[12] 則是利用分布假說並加上詞庫中的資訊來訓練詞向

量。

理論上，這類從頭訓練詞向量的方法，雖然需要較長的訓練時間，但其在訓練過程中應能更有效的學習文本與詞庫中的字詞關係的知識，比起「調整預訓練詞向量」這類方法應能獲得更好的詞向量。但根據實驗過往研究發現，其所產生的詞向量不如「調整預訓練詞向量」的方法¹。

2.2 使用詞庫中的字詞關係調整預訓練詞向量

本節介紹藉由詞庫中已知的知識來調整預訓練的詞向量。此類方法不需要使用大量文本訓練出新的詞向量，可透過其他附加知識來調整詞向量，除可訓練出較符合詞庫知識的詞向量，更可節省大量的訓練時間。

2.2.1 Retrofitting Word Vectors to Semantic Lexicons

Retrofitting[14] 於 2015 年被提出。此論文使用其他學者預訓練完成的詞向量，例如 GloVe，再加上詞庫中萃取出的同義詞關係，訓練出的新的詞向量。其目標是將詞庫中的同義詞對在向量空間中的距離拉近，其損失函數如下：

$$\Psi(\hat{Q}) = \sum_{i=1}^n \left[\alpha_i \|\hat{q}_i - q_i\|^2 + \sum_{(i,j) \in E} \beta_{ij} \|\hat{q}_i - \hat{q}_j\|^2 \right] \quad (2.1)$$

上式中， $Q = (q_1, \dots, q_n)$ 、 $\hat{Q} = (\hat{q}_1, \dots, \hat{q}_n)$ 分別表示調整前後的詞向量。 E 為詞庫中萃取出的同義詞關係資料集， j 為 i 同義詞。前式目標為控制調整前後 i 的詞向量在向量空間的差異，而後者的目標則是利用歐幾里得距離拉近同義詞對在向量空間的距離。另外， α 及 β 分別為前後式的超參數。

另外，為了增加訓練資料，Retrofitting 對 PPDB 資料集進行更進一

¹SimLex-999 官網 (www.cl.cam.ac.uk/fh295/simlex.html) 條列對其資料集表現最後之模型。

步的處理。由於 PPDB 含有多種不同語言，若兩個不同的詞在翻譯成另一個語言後為同義詞，則也將他們視為同義詞。

2.2.2 ATTRACT-REPEL

ATTRACT-REPEL[17] 的目標是要藉由詞庫中的同義詞反義詞資訊來調整預訓練的詞向量。另外，ATTRACT-REPEL 除了調整英文的詞向量，更在其他 15 種語言上進行實驗。此研究更將不同語言訓練成同一個空間上的詞向量，藉此讓一些文本與詞庫資源較少的語言能獲得良好的訓練。

本研究的目標是以餘弦距離將同義詞拉近、反義詞拉遠。假設也一詞對 (x_l, x_r) ，其所對應的詞向量為 $(\mathbf{x}_l, \mathbf{x}_r)$ 。而優化過程的訓練集大小 (mini-batch) 定義為 B_S (大小為 k_1) 及 B_A (大小為 k_2)，分別表示同反義詞的訓練集大小。此外，將 $T_S(B_S) = [(t_l^1, t_r^1), \dots, (t_l^{k_1}, t_r^{k_1})]$ 與 $T_A(B_A) = [(t_l^1, t_r^1), \dots, (t_l^{k_2}, t_r^{k_2})]$ 分別定義為同義詞對與反義詞對的 negative examples。其損失函數如下：

$$S(B_S) = \sum_{(x_l, x_r) \in B_S} [\tau(\delta_{syn} + \mathbf{x}_l \mathbf{t}_l - \mathbf{x}_l \mathbf{x}_r) + \tau(\delta_{syn} + \mathbf{x}_r \mathbf{t}_r - \mathbf{x}_l \mathbf{x}_r)] \quad (2.2)$$

$$A(B_A) = \sum_{(x_l, x_r) \in B_A} [\tau(\delta_{ant} + \mathbf{x}_l \mathbf{x}_r - \mathbf{x}_l \mathbf{t}_l) + \tau(\delta_{ant} + \mathbf{x}_l \mathbf{x}_r - \mathbf{x}_r \mathbf{t}_r)] \quad (2.3)$$

式 2.2、2.3 中， $\tau(x) = \max(0, x)$ 是取其最大值， δ_{syn} 為控制同義詞要多靠近彼此的超參數， δ_{ant} 則為控制反義詞要多遠離彼此的超參數。兩式目標是將同義詞對拉近，並使其在項量空間上的距離比 negative examples 更近；反之反義詞對的目標則是拉遠，並使其在項量空間上的

距離比 negative examples 更遠。

$$R(B_S, B_A) = \sum_{\mathbf{x}_i \in V(B_S \cup B_A)} \lambda_{reg} \|\hat{\mathbf{x}}_i - \mathbf{x}_i\|_2 \quad (2.4)$$

式 2.4 的目標則是控制同反義詞對被拉的距離，使其不與調整前的詞向量相距過遠。 λ_{reg} 為 L2 正規化 (L2 Regularisation)， $\hat{\mathbf{x}}_i$ 為調整前的詞詞向量。最終損失函數定義為三式加總，如式 2.5。

$$C(B_S, B_A) = S(B_S) + A(B_A) + R(B_S, B_A) \quad (2.5)$$

2.2.3 自注意力機制

此論文 [4] 提出利用自注意力機制來讓模型學習以 listwise 形式做為輸入的詞類關係。由於同義詞反義詞為無序列關係的集合，因此此方法可讓字詞同時學習到其他輸入的字詞。其目標是將詞庫中的同義詞在向量空間中的距離拉近、反義詞則拉遠，其損失函數如下：

$$l_1 = \frac{\sum_{s \in S_i} (1 - \cos(w'_i, s')) + \sum_{a \in A_i} \cos(w'_i, a')}{n + m} \quad (2.6)$$

$$l_2 = \frac{\text{mse}(w_i, w'_i) + \sum_{s \in S_i} \text{mse}(s, s') + \sum_{a \in A_i} \text{mse}(a, a')}{1 + n + m} \quad (2.7)$$

$$\text{Loss} = \alpha l_1 + (1 - \alpha) l_2 \quad (2.8)$$

其中 w_i 及 w'_i 為分別為調整前後的詞向量。 S_i 為 w_i 的同義詞集合， n 為 S_i 的大小； A_i 為 w_i 的反義詞集合， m 為 A_i 的大小。

式 2.6 中， \cos 為餘弦相似度 (cosine similarity)，其目標是利用餘弦相似度來分別拉近及拉遠同反義詞。式 2.7 中， mse 則為均方差 (mean-square error)，其目標是要控制詞向量在調整前後的差異。式 2.8 則是要

控制兩式調整程度， α 為其超參數。

二、相關研究

三、 研究模型及方法

3.1 Task 1：調整詞向量

3.1.1 資料前處理

本研究目標是將同義詞與反義詞同時加入模型進行訓練，如此一個字詞才能同時學習到其他所以同反義詞資訊，其同反義詞也能學習到彼此間的資訊。而若要使一個字詞能同時學習到其他所以字詞的資訊，則需無方向性的模型，如圖 3.1所示。

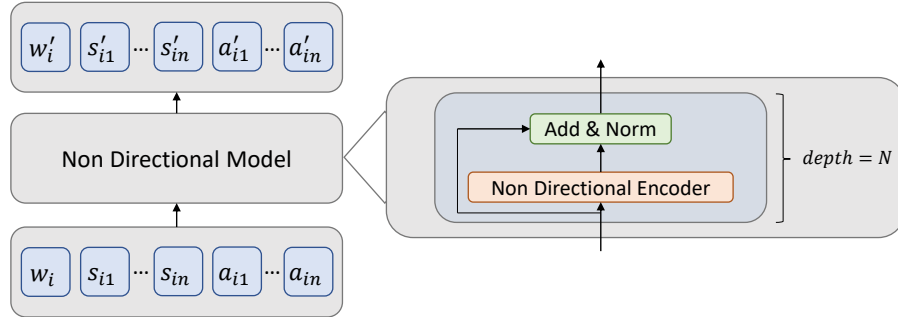


圖 3.1: 無方向性模型架構

因此，我們先對同反義詞資料集進行處理。若以 W 表示為預訓練詞向量，對於 $w_i \in W$ ，若資料集中有 w_i 的同義詞集合 $S_i = \{s_{i1}, \dots, s_{in}\}$ 與反義詞集合 $A_i = \{a_{i1}, \dots, a_{im}\}$ ，我們會將 S_i 及 A_i 接在 w_i 之後，將其處理成序列的形式做為模型輸入，即 $input = \{w_i, s_{i1}, \dots, s_{in}, a_{i1}, \dots, a_{im}\}$ 。並經過多層無方向性的序列模型訓練之後，我們會得到新的詞向量。

3.1.2 自注意力機制 (Self-Attention Mechanism)

由於本研究做為輸入的序列 $input = \{w_i, s_{i1}, \dots, s_{in}, a_{i1}, \dots, a_{im}\}$ ，序列中的每個字詞並沒有順序性，且我們希望輸入序列中的所有字都能互相學習到彼此的資訊，因此我們選擇不使用一般常用在自然語言處理的自迴歸模型 (Autoregressive Model)[20] 或循環神經網路 (Recurrent Neural Network)[21], [22]，我們借鏡 [4] 選擇以無方向性的方法來做為我們的訓練模型，也就是 [23] 所提出的自注意力機制。

自注意力機制的目標是想取代循環神經網路所能做的事情。自注意力機制與雙向的循環神經網路相同的是兩者的輸入序列中的每個單位都能學習到輸入序列中的其他單位，但不同的是雙向的循環神經網路是計算完一個輸出之後才能再算下一個，而自注意力機制則是可以同時處理整個輸入序列並輸出。

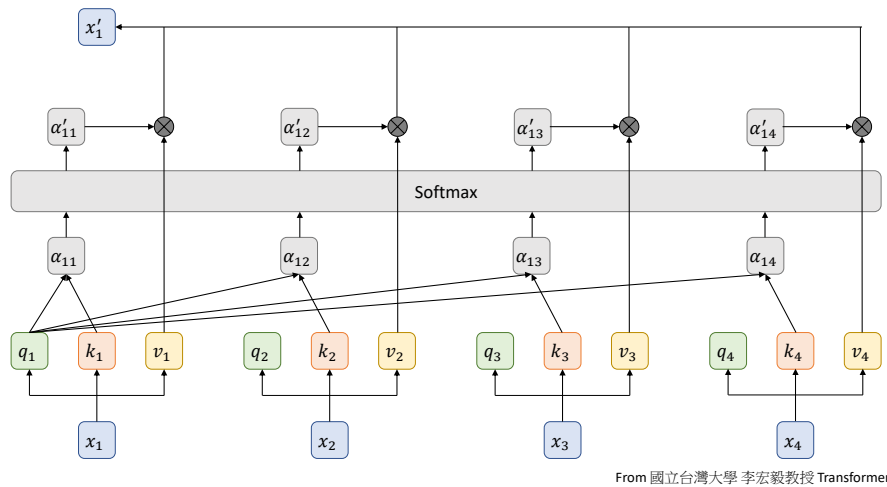


圖 3.2: 自注意力機制計算流程

如圖 3.2，假設一輸入序列為 x_1, \dots, x_4 ，每個輸入會分別乘上權重 W_q 、 W_k 、 W_v 轉換為 query、key、value，而 W_q 、 W_k 、 W_v 為透過訓練產生的參數。以 x_1 為例，轉換後產生 q_1 、 k_1 、 v_1 。之後， q_1 會與自己和其他單位的 key(k_1, \dots, k_4) 分別做內積得到分數 $\alpha(\alpha_{11}, \dots, \alpha_{14})$ ，再經過 Softmax 做正規化，得到 $\alpha'(\alpha'_{11}, \dots, \alpha'_{14})$ 。完成正規化之後的再與各個

$\text{value}(v_1, \dots, v_4)$ 做內積，即為最後的輸出 x'_1 。其運算式如下：

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (3.1)$$

式中， Q 、 K 、 V 是每個輸入的 query、key、value 集合而成的矩陣。 d_k 為 key 的維度，其目的是防止 query 及 key 做內積過後分數過大。

3.1.3 損失函數

本研究使用三種損失函數進行調整，我們將三者分別定義為 $Loss_{Cos+Eu}$ 、 $Loss_{Eu+Eu}$ 、 $Loss_{Eu+MSE}$ ，並分別訓練出三種不同的詞向量。其中，有兩種損失函數與 [14] 同樣使用歐幾里得距離為基礎計算；另一種則是與 [4] 一樣使用餘弦距離，不過稍有調整。之所以使用不同的損失函數進行計算是因為我們在實驗過程當中發現，相似的損失函數所帶來的結果卻有極大的差距，例如歐幾里得距離與均方差 (mean-square error) 差別只在於有無取平方根，其所帶來的結果卻相當不一樣。另外，我們也分別使用了一般在詞向量領域常見的餘弦距離與不常見的歐幾里得距離進行計算。

式 3.2 及 3.3 目標為將字詞與其同義詞拉近、反義詞拉遠，兩式分別使用餘弦距離與歐幾里得距離進行計算，前者運用在 $Loss_{Cos+Eu}$ 上，而後者運用在 $Loss_{Eu+Eu}$ 及 $Loss_{Eu+MSE}$ 上。而式 3.4 及式 3.5 目標分別為使用歐幾里得距離及均方差控制同一個字詞在調整前後的移動程度，前者運用在 $Loss_{Cos+Eu}$ 及 $Loss_{Eu+Eu}$ 上，而後者運用在 $Loss_{Eu+MSE}$ 上。四式如下：

$$l_{1_{Cos}} = \frac{\sum_{s_j \in S_i} (1 - \cos(w'_i, s'_j)) + \sum_{a_j \in A_i} \cos(w'_i, a'_j)}{n + m} \quad (3.2)$$

$$l_{1_{Eu}} = \frac{\sum_{s_j \in S_i} |w'_i - s'_j| - \sum_{a_j \in A_i} |w'_i - a'_j|}{n + m} \quad (3.3)$$

$$l_{2_{Eu}} = \frac{|w'_i - w_i| + \sum_{s_j \in S_i} |s'_j - s_j| + \sum_{a_j \in A_i} |a'_j - a_j|}{1 + n + m} \quad (3.4)$$

$$l_{2_{MSE}} = \frac{\text{mse}(w_i, w'_i) + \sum_{s_j \in S_i} \text{mse}(s_j, s'_j) + \sum_{a_j \in A_i} \text{mse}(a_j, a'_j)}{1 + n + m} \quad (3.5)$$

其中, w_i 、 s_j 、 a_j 為調整前的詞向量, w'_i 、 s'_j 、 a'_j 則為調整後的詞向量。 S_i 及 A_i 分別為 w_i 的同義詞集合與反義詞集合, n 、 m 分別為 S_i 及 A_i 的大小。

$$Loss_{Cos+Eu} = \alpha l_{1_{Cos}} + (1 - \alpha) l_{2_{Eu}} \quad (3.6)$$

$$Loss_{Eu+Eu} = \beta l_{1_{Eu}} + (1 - \beta) l_{2_{Eu}} \quad (3.7)$$

$$Loss_{Eu+MSE} = \gamma l_{1_{Eu}} + (1 - \gamma) l_{2_{MSE}} \quad (3.8)$$

式 3.6 及 3.7 中, α 、 β 、 γ 分別為三式的超參數, 其作用為控制三式的調整程度。

3.2 Task 2：自然語言處理任務

本階段實驗目標為透過下游的自然語言處理任務, 比較經上階段實驗調整後的詞向量是否較調整前獲得更多的語義資訊。因此我們選擇使用較簡單的自然語言處理模型, 而非目前自然語言處理領域中最為先進的模型, 如 XLNet[24] 及 Bert[25]。我們有嘗試過使用 XLNet 進行實驗, 但其模型不管使用何種詞向量進行下游任務的實驗比較、甚至使用隨機初始化的詞向量進行, 其結果均能達到幾乎一樣的水準, 無法比較何種

詞向量具有較好的語義資訊。

3.2.1 長短期記憶模型 (LSTM)

由於上面提到的因素，我們選擇使用長短期記憶模型 (LSTM)[21]。而本研究所進行的下游自然語言處理任務—IMDb[26] 及 Twitter US Airline Sentiment¹，皆為判斷一個敘述句為正面評論或反面評論，因此我們選擇使用對有順序性資料有良好訓練能力的長短期記憶模型。另外，我們使用 binary cross-entropy(BCE) 做為損失函數，其所搭配的激勵函數則為 sigmoid，其式如下：

$$BCE = -\frac{1}{N} \sum_{i=1}^N y_i \log(p(y_i)) + (1 - y_i) \log(1 - p(y_i)) \quad (3.9)$$

式 3.9 中， y_i 為資料正確的標籤 (true label)， $p(y_i)$ 則是模型的預測， i 則代表第 i 筆資料，共 N 筆資料。

¹<https://www.kaggle.com/crowdflower/twitter-airline-sentiment>

三、研究模型及方法

四、實驗結果與分析

4.1 Task 1：調整詞向量

4.1.1 模型設置

透過 [4] 所做的實驗，我們以單頭式的自注意力機制模型做為實驗模型。另外，理論上此模型輸入的序列長度應越長越好，如此才能讓一個字詞學到更多與其他字的資訊，但透過實驗，我們發現若將序列長度設定為同反義詞集合中最長的兩筆資料總和等長，我們設備的顯存無法裝下如此巨大的神經網路，即使解決硬體問題，其所需的訓練時間也無法展現出本實驗方法的優勢。因此我們同 [4] 論文中的實驗，將輸入的序列長度設置為第三四分位數 (Q_3)。

關於超參數 α 、 β 、 γ 的設定，我們取 $\alpha, \beta, \gamma \in [0.1, 0.2, \dots, 0.9]$ ，並實驗取其最好的結果作為最終超參數的值。在本篇論文作為比較的方法 [4]，經實驗我們將其超參數定為 0.3；損失函數 $Loss_{Cos+Eu}$ 定為 0.1；損失函數 $Loss_{Eu+Eu}$ 定為 0.3；損失函數 $Loss_{Eu+MSE}$ 定為 0.5。

另外，不同於其他相關研究以相似度分數資料集做為測試資料，我們選擇將每個同義詞反義詞資料集分成訓練資料與測試資料，分別佔 70% 與 30%。因為他們所使用的相似度分數資料集所給定的相似度分數不一定是依照其同反義詞關係給出的，可能有其他關聯使之分數提高，另外，其中的反義詞與無關聯的詞分數也很接近，無從區分。

我們將評量方式分為兩種，一種是以相似度任務來評估，本研究計

算模型輸出的詞向量與資料集內人工標註的相似度分數之間的斯皮爾曼等級相關係數 (Spearman's rank correlation); 另一種是計算測試資料裡的同反義詞在調整前後的餘弦距離計算歐幾里得距離。兩式如下:

$$\text{loss}_{\text{Cosine}} = \frac{\sum_{s_j \in S_i} (1 - \cos(w'_i, s'_j)) + \sum_{a_j \in A_i} \cos(w'_i, a'_j)}{n + m} \quad (4.1)$$

$$\text{loss}_{\text{Euclidean}} = \frac{\sum_{s_j \in S_i} |w'_i - s'_j| + \sum_{a_j \in A_i} |w'_i - a'_j|}{n + m} \quad (4.2)$$

4.1.2 實驗資料集

本節介紹本研究進行詞向量調整所用到的資料集，其中包括預訓練的詞向量，及用來調整詞向量的同義詞反義詞資料集。除了上述兩種訓練詞向量的資料集外，還有進行模型評估的相似度分數資料集。

4.1.2.1 預訓練詞向量

GloVe[2] 為史丹佛大學所訓練的詞向量，其思路與 Word2Vec[3] 類似，但 GloVe 充分考慮了詞的共現情況。本研究使用兩種不同的 GloVe，分別是以 Wikipedia 2014 與 Gigaword 5 等文本訓練而得的 GloVe_{6B} 及以 Common Crawl 文本訓練而得的 GloVe_{42B}，兩者差別在於用來訓練的文本大小，皆可從其網站¹ 取得。

Word2Vec[3] 為 Google 於 2013 一款計算低維詞向量的工具，其從 NNLM[27] 改良而來。Word2Vec 透過學習上下文資訊訓練而得的詞向量。本研究所使用的 Word2Vec 是 [28] 以 Skip-Gram 為模型訓練 Wikipedia 文本而得，可從其網站² 取得。

¹<https://nlp.stanford.edu/projects/glove/>

²<https://wikipedia2vec.github.io/wikipedia2vec/pretrained>

fastText[29] 可視為 Word2Vec[3] 的擴充。其以 Word2Vec 模型為基礎，加上 sub-word 資訊，使詞向量能夠學習到更豐富的資訊。本研究所使用的 fastText 是以 Wikipedia 2017、UMBC webbase corpus、statmt.org news dataset 等資料集訓練而得，可從其網站³取得。

4.1.2.2 同義詞、反義詞資料集

PPDB[30] 是一個包含 16 種語言、上百萬種釋義 (paraphrases) 的資料集。其產生的目的是為了在機器學習語言的過程中，使其能夠學習到文字多樣的變化與不常見的字。

我們根據 PPDB2.0 所新增的字詞的特徵關定一同反義詞。另外，我們利用 Retrofitting[14] 中所使用的方法，增加同義詞的量。透過以上兩種方法，我們得到同義詞資料集 $PPDB_{syn}$ ，及加入了反義詞的資料集 $PPDB_{syn+ant}$ 。

此外，我們也使用 [31] 的方法從 PPDB 提取出更多的反義詞，產生新的同反義詞資料集 $PPDB_{syn+ant+}$ 。其中， $PPDB_{syn+ant}$ 與 $PPDB_{syn+ant+}$ 所含的反義詞分別佔 1% 與 44%。

WordNet[32] 是一個英文的語意資料集，其包含同義詞集 (synsets)，我們從此同義詞集中取得同義詞資料集 $WordNet_{syn}$ 。另外，再加上反義詞資訊產生同反義詞資料集 $WordNet_{syn+ant}$ 。

FrameNet[33] 是一個英文的語意資料集，其包含超過 20 萬筆標記過的句子及其所對應的超過 1200 個語意框架，其是一個可供語意關係標記的資料集。我們同樣借 Retrofitting[14] 所使用的方法，產生出同義詞資料集 $FrameNet_{syn}$ 。

³<https://fasttext.cc/docs/en/english-vectors.html>

Lexicon 此為上述三類同反義詞資料集的集合，產生此資料集的目的是運用大量同反義詞資料進行訓練是否有較好結果。共分為三類，包含只有同義詞的資料集 $\text{Lexicon}_{\text{syn}}$ 與同時含有同反義詞的 $\text{Lexicon}_{\text{syn+ant}}$ 及 $\text{Lexicon}_{\text{syn+ant}^+}$ 。其中 $\text{Lexicon}_{\text{syn+ant}}$ 和 $\text{Lexicon}_{\text{syn+ant}^+}$ 差別為一個所包含的是 $\text{PPDB}_{\text{syn+ant}}$ ，另一個則是 $\text{PPDB}_{\text{syn+ant}^+}$ 。

4.1.2.3 相似度分數資料集

本研究所使用相似度分數資料集為 **SimLex-999**[34]，此資料集含有 999 組詞對及 0~10 分的相似度分數。相較於其他相似度分數資料集，SL-999 在詞對若為反義詞所給的分數，與兩個不相關詞組成的詞對所給的分數較有差異，因此我們使用此資料集進行相似度任務的評估。

4.1.3 實驗結果

4.1.3.1 使用不同同反義詞資料集調整預訓練詞向量 GloVe

理論上，當所使用的訓練資料越多，訓練出來的結果會越好。 GloVe_{6B} 與 GloVe_{42B} 的差別即是他們所使用的訓練資料多寡，我們也可以從表 4.1~4.5 發現 GloVe_{42B} 表現確實比 GloVe_{6B} 來得好。因此本實驗希望透過對 GloVe_{6B} 進行調整，看看調整後的詞向量在測試資料集的表現是否較調整前好，甚至接近 GloVe_{42B} 的表現。

與 GloVe_{6B} 相比， GloVe_{42B} 使用更大量的文本進行訓練，但同時其所需要的訓練時間也相對較多。

另外，我們也可以從表 4.2~4.5 發現，以 [4] 提出的損失函數 $\text{Loss}_{\text{Cos}+MSE}$ 或本研究所提出的損失函數 $\text{Loss}_{\text{Eu}+MSE}$ 進行調整時，若使用特定同義詞反義詞資料集對 GloVe_{6B} 進行調整，甚至可以使其表現得比調整前的 GloVe_{42B} 來得好。另外，以本研究所提出的另外兩種損失函數來調整的詞向量，我們可以從表中看到以 $\text{Loss}_{\text{Cos}+Eu}$ 調整的詞向量表現均較調整前差，而 $\text{Loss}_{\text{Eu}+Eu}$ 雖然表現沒有 [4] 來得好，但大部分情況下仍然有進

表 4.1: 使用 FrameNet_{syn} 資料集調整預訓練詞向量 GloVe。數據標示部分，粗體為最佳數據。

FrameNet _{syn} 調整	Loss Function	$\text{loss}_{\text{Cosine}}$	$\text{loss}_{\text{Euclidean}}$	同義詞 拉近比例
GloVe _{6B}	Original	0.8122	1.2659	
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.7524 ± 0.0015	1.2164 ± 0.0012	82 ± 1.3%
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.8733 ± 0.0017	1.3144 ± 0.0012	42 ± 0.7%
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.8096 ± 0.0018	1.2662 ± 0.0014	59 ± 3.0%
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.7896 ± 0.0020	1.2489 ± 0.0016	61 ± 1.4%
GloVe _{42B}	Original	0.7122	1.1828	
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.6163 ± 0.0060	1.0987 ± 0.0054	86 ± 1.9%
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.7306 ± 0.0025	1.1995 ± 0.0021	22 ± 2.5%
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.7126 ± 0.0016	1.1839 ± 0.0013	54 ± 2.2%
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.6408 ± 0.0075	1.1170 ± 0.0068	69 ± 2.4%

表 4.2: 使用 PPDB_{syn} 資料集調整預訓練詞向量 GloVe。數據標示部分，粗體為最佳數據。

	Loss Function	$\text{loss}_{\text{Cosine}}$	$\text{loss}_{\text{Euclidean}}$	同義詞 拉近比例
GloVe _{6B}	Original	0.7892	1.2455	
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.6581 ± 0.0004	1.1343 ± 0.0004	96 ± 0.0%
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.7970 ± 0.0004	1.2525 ± 0.0002	26 ± 0.6%
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.7798 ± 0.0002	1.2384 ± 0.0001	75 ± 0.4%
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.6203 ± 0.0026	1.1008 ± 0.0027	92 ± 0.1%
GloVe _{42B}	Original	0.6724	1.1476	
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.5096 ± 0.0004	0.9953 ± 0.0003	99 ± 0.0%
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.6871 ± 0.0001	1.1611 ± 0.0000	19 ± 0.3%
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.6485 ± 0.0002	1.1272 ± 0.0003	87 ± 0.3%
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.3760 ± 0.0024	0.8520 ± 0.0030	98 ± 0.1%

步。除此之外，表中的「同義詞拉近比例」與「反義詞拉近比例」欄位分別代表測試資料集中同義詞拉近的數量百分比與反義詞拉遠的數量百分比，有些表格無「反義詞拉近比例」欄位是因為該用來調整同反義詞的資料集只有同義詞。我們可以發現，雖然反義詞的表現很差，但相反地，幾乎所有同義詞都有確實被拉進。因此我們推測同義詞相較於反義詞在

表 4.3: 使用 PPDB_{syn+ant} 資料集調整預訓練詞向量 GloVe。數據標示部分，粗體為最佳數據。

	Loss Function	$\text{loss}_{\text{Cosine}}$	$\text{loss}_{\text{Euclidean}}$	同義詞 拉近比例	反義詞 拉遠比例
GloVe _{6B}	Original	0.7865	1.2337		
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.6564 ± 0.0003	1.1233 ± 0.0003	$96 \pm 0.1\%$	$1 \pm 0.4\%$
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.7938 ± 0.0001	1.2402 ± 0.0001	$28 \pm 0.6\%$	$70 \pm 3.0\%$
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.7772 ± 0.0001	1.2266 ± 0.0001	$75 \pm 0.2\%$	$34 \pm 3.3\%$
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.6213 ± 0.0016	1.0923 ± 0.0015	$91 \pm 0.1\%$	$0 \pm 0.1\%$
GloVe _{42B}	Original	0.6711	1.1369		
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.5095 ± 0.0007	0.9859 ± 0.0008	$99 \pm 0.0\%$	$0 \pm 0.1\%$
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.6843 ± 0.0004	1.1490 ± 0.0002	$20 \pm 0.4\%$	$78 \pm 4.2\%$
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.6469 ± 0.0009	1.1163 ± 0.0008	$88 \pm 0.5\%$	$22 \pm 2.3\%$
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.3758 ± 0.0033	0.8420 ± 0.0040	$98 \pm 0.1\%$	$0 \pm 0.0\%$

表 4.4: 使用 WordNet_{syn} 資料集調整預訓練詞向量 GloVe。數據標示部分，粗體為最佳數據。

	Loss Function	$\text{loss}_{\text{Cosine}}$	$\text{loss}_{\text{Euclidean}}$	同義詞 拉近比例
GloVe _{6B}	Original	0.8302	1.2805	
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.7372 ± 0.0002	1.2034 ± 0.0002	$96 \pm 0.1\%$
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.8340 ± 0.0001	1.2845 ± 0.0001	$50 \pm 0.3\%$
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.8313 ± 0.0002	1.2833 ± 0.0002	$54 \pm 0.4\%$
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.6941 ± 0.0008	1.1666 ± 0.0007	$92 \pm 0.1\%$
GloVe _{42B}	Original	0.7169	1.1871	
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.5570 ± 0.0005	1.0421 ± 0.0005	$99 \pm 0.0\%$
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.7272 ± 0.0002	1.1963 ± 0.0001	$24 \pm 0.5\%$
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.6986 ± 0.0002	1.1718 ± 0.0002	$86 \pm 0.5\%$
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.4372 ± 0.0028	0.9209 ± 0.0030	$98 \pm 0.1\%$

調整詞向量上是更有用的資訊。

表 4.5: 使用 WordNet_{syn+ant} 資料集調整預訓練詞向量 GloVe。數據標示部分，粗體為最佳數據。

	Loss Function	$\text{loss}_{\text{Cosine}}$	$\text{loss}_{\text{Euclidean}}$	同義詞 拉近比例	反義詞 拉遠比例
GloVe _{6B}	Original	0.8273	1.2696		
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.7349 ± 0.0003	1.1931 ± 0.0003	$96 \pm 0.1\%$	$0 \pm 0.0\%$
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.8311 ± 0.0002	1.2737 ± 0.0001	$50 \pm 0.1\%$	$83 \pm 1\%$
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.8281 ± 0.0002	1.2722 ± 0.0001	$55 \pm 0.2\%$	$35 \pm 1.0\%$
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.6896 ± 0.0017	1.1543 ± 0.0015	$92 \pm 0.1\%$	$0 \pm 0.0\%$
GloVe _{42B}	Original	0.7151	1.1171		
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.5568 ± 0.0005	1.0336 ± 0.0005	$99 \pm 0.0\%$	$0 \pm 0.0\%$
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.7252 ± 0.0003	1.1861 ± 0.0002	$25 \pm 0.5\%$	$85 \pm 4.3\%$
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.6950 ± 0.0004	1.1604 ± 0.0003	$86 \pm 0.5\%$	$16 \pm 5.4\%$
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.4358 ± 0.0015	0.9113 ± 0.0017	$98 \pm 0.1\%$	$0 \pm 0.0\%$

4.1.3.2 使用不同同反義詞資料集調整其他預訓練詞向量

上一節，我們說明了詞向量 GloVe 被調整的結果。而本節我們使用不同的同義詞反義詞資料集對其他的預訓練詞向量進行調整，並比較調整前後的詞向量在測試資料中的表現。

從表 4.6 及 4.7 我們可以發現，除了 $\text{Loss}_{\text{Cos}+\text{Eu}}$ 調整的詞向量，不管使用何種資料集、何種損失函數來調整詞向量，幾乎都能使其有更好的表現。我們也可以發現，相較於幾乎都有被調整的同義詞，反義詞其實沒有被調整到，如上一節推測，也許同義詞相較之下是更有用的資訊，不過這有待更進一步的實驗證明。另外，使用 FrameNet_{syn} 資料集調整時，進步幅度沒有其他的資料集來的那麼高、甚至更差，正如相關研究所做的推測，相關研究顯示「調整後詞向量的表現與訓練資料量成正相關」，也因此我們更進一步地做了下一節的實驗以驗證此兩項推測。

除此之外，值得注意地是表 4.7。相較於 Glove 及 Word2Vec 這兩種詞向量，fastText 本身在預訓練時就已經加入更豐富的資訊一起做訓練，我們原本認為其進步幅度不會叫前兩者來的顯著，但實際上 fastText 在調整之後的進步幅度更甚 Glove 及 Word2Vec。

表 4.6: 使用不同同反義詞資料集調整預訓練詞向量 Word2Vec。數據標示部分，粗體為最佳數據。

	Loss Function	$\text{loss}_{\text{Cosine}}$	$\text{loss}_{\text{Euclidean}}$	同義詞 拉近比例	反義詞 拉遠比例
FrameNet _{syn}	Original	0.6834	1.1620		
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.7889 ± 0.0027	1.2471 ± 0.0020	$23 \pm 0.5\%$	
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.7191 ± 0.0009	1.1936 ± 0.0007	$7 \pm 1.0\%$	
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.6925 ± 0.0021	1.1708 ± 0.0019	$47 \pm 2.2\%$	
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.7045 ± 0.0016	1.1758 ± 0.0014	$47 \pm 0.4\%$	
PPDB _{syn}	Original	0.671	1.1488		
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.3808 ± 0.0003	0.8627 ± 0.0003	$99 \pm 0.0\%$	
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.6979 ± 0.0001	1.1727 ± 0.0001	$1 \pm 0.0\%$	
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.6397 ± 0.0006	1.1220 ± 0.0005	$92 \pm 0.3\%$	
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.3237 ± 0.0046	0.7849 ± 0.0060	$98 \pm 0.1\%$	
PPDB _{syn+ant}	Original	0.6694	1.1378		
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.3819 ± 0.0004	0.8544 ± 0.0005	$99 \pm 0.0\%$	$0 \pm 0.0\%$
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.6961 ± 0.0002	1.1614 ± 0.0001	$1 \pm 0.0\%$	$99 \pm 0.2\%$
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.6376 ± 0.0007	1.1106 ± 0.0006	$93 \pm 0.3\%$	$0 \pm 0.0\%$
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.3269 ± 0.0055	0.7792 ± 0.0073	$98 \pm 0.0\%$	$1 \pm 0.2\%$
WordNet _{syn}	Original	0.7071	1.1824		
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.4255 ± 0.0002	0.9147 ± 0.0002	$99 \pm 0.0\%$	
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.7264 ± 0.0001	1.1990 ± 0.0001	$3 \pm 0.2\%$	
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.6807 ± 0.0006	1.1601 ± 0.0005	$93 \pm 0.5\%$	
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.5688 ± 0.0055	1.046 ± 0.0054	$76 \pm 0.6\%$	
WordNet _{syn+ant}	Original	0.7052	1.1723		
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.6898 ± 0.0005	1.1552 ± 0.0004	$55 \pm 0.1\%$	$30 \pm 1.0\%$
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.7242 ± 0.0003	1.1887 ± 0.0002	$3 \pm 0.1\%$	$97 \pm 2.1\%$
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.6787 ± 0.0004	1.1499 ± 0.0004	$93 \pm 0.2\%$	$0 \pm 0.0\%$
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.5679 ± 0.0028	1.0390 ± 0.0026	$76 \pm 0.5\%$	$8 \pm 0.8\%$

表 4.7: 使用不同同反義詞資料集調整預訓練詞向量 fastText。數據標示部分，粗體為最佳數據。

	Loss Function	$\text{loss}_{\text{Cosine}}$	$\text{loss}_{\text{Euclidean}}$	同義詞 拉近比例	反義詞 拉遠比例
FrameNet _{syn}	Original	0.5730	1.0634		
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.5018 ± 0.0123	0.9524 ± 0.0122	57 ± 0.6%	
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.7811 ± 0.0043	1.2437 ± 0.0032	3 ± 0.4%	
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.5375 ± 0.0055	0.9850 ± 0.0042	53 ± 0.4%	
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.8492 ± 0.0073	1.2962 ± 0.0056	6 ± 0.5%	
PPDB _{syn}	Original	0.5529	1.0403		
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.0701 ± 0.0002	0.3473 ± 0.0006	99 ± 0.0%	
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.5842 ± 0.0002	1.0706 ± 0.0002	1 ± 0.0%	
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.5934 ± 0.0004	1.0792 ± 0.0004	0 ± 0.0%	
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.4398 ± 0.0255	0.9211 ± 0.0283	77 ± 4.4%	
PPDB _{syn+ant}	Original	0.5526	1.0306		
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.0740 ± 0.0001	0.3435 ± 0.0003	99 ± 0.0%	0 ± 0.0%
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.5839 ± 0.0001	1.0608 ± 0.0001	1 ± 0.0%	98 ± 1.6%
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.5935 ± 0.0008	1.0694 ± 0.0015	0 ± 0.0%	0 ± 0.0%
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.4283 ± 0.0260	0.8989 ± 0.0289	79 ± 4.1%	26 ± 7.2%
WordNet _{syn}	Original	0.5706	1.0595		
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.0774 ± 0.0003	0.3802 ± 0.0009	99 ± 0.0%	
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.7813 ± 0.0009	1.2430 ± 0.0007	2 ± 0.1%	
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.6024 ± 0.0029	1.0896 ± 0.0027	1 ± 0.5%	
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.5403 ± 0.0133	1.0260 ± 0.0105	62 ± 3.9%	
WordNet _{syn+ant}	Original	0.5700	1.0505		
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.0807 ± 0.0003	0.3762 ± 0.0009	99 ± 0.0%	0 ± 0.0%
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.7785 ± 0.0002	1.2322 ± 0.0002	2 ± 0.1%	99 ± 0.0%
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.6004 ± 0.0019	1.0794 ± 0.0017	1 ± 0.2%	0 ± 0.0%
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.5441 ± 0.0157	1.0253 ± 0.0152	61 ± 5.3%	16 ± 6.5%

4.1.3.3 使用含有較多同反義詞的資料集調整不同的預訓練詞向量

本節實驗我們要驗證相關研究所發現的論述「調整後詞向量的表現與訓練資料量成正相關」及前面實驗提到的「同義詞相較於反義詞在調整詞向量上是更有用的資訊」是否正確。因此我們使用含有較多資料的 PPDB 及 Lexicon 進行實驗。

我們先來檢視 PPDB 及 Lexicon 兩種資料集的調整結果。如表 4.8~4.15 所顯示，調整後進步幅度最多的反而是使用 PPDB_{syn} 及 Lexicon_{syn} 調整的詞向量。也確實證明前述提到的「同義詞相較於反義詞在調整詞向量上是更有用的資訊」。

接著，我們從表 4.12~4.15 比對表 4.8~4.11 (兩者的 syn、syn+ant、syn+ant⁺ 分別比對) 發現，使用 Lexicon 調整的詞向量進步幅度均優於使用 PPDB 調整的詞向量。也就是說，當使用的訓練資料越多，調整後的進步幅度越大，也證實了相關研究所提出的「調整後詞向量的表現與訓練資料量成正相關」的論述。

表 4.8: 使用不同 PPDB 資料集調整預訓練詞向量 GloVe_{6B}。數據標示部分，粗體為最佳數據。

	Loss Function	loss_{Cosine} (vs. Original)	loss_{Euclidean} (vs. Original)
PPDB _{syn}	Original	0.7892	1.2455
	Lin[4](<i>Loss_{Cos+MSE}</i>)	0.6581 ± 0.0004(-0.1311)	1.1343 ± 0.0004(-0.1112)
	<i>Loss_{Cos+Eu}</i>	0.7970 ± 0.0004(+0.0078)	1.2525 ± 0.0002(+0.0070)
	<i>Loss_{Eu+Eu}</i>	0.7798 ± 0.0002(-0.0094)	1.2384 ± 0.0001(-0.0071)
	<i>Loss_{Eu+MSE}</i>	0.6203 ± 0.0026(-0.1689)	1.1008 ± 0.0027(-0.1447)
PPDB _{syn+ant}	Original	0.7865	1.2337
	Lin[4](<i>Loss_{Cos+MSE}</i>)	0.6564 ± 0.0003(-0.1301)	1.1233 ± 0.0003(-0.1104)
	<i>Loss_{Cos+Eu}</i>	0.7938 ± 0.0001(+0.0073)	1.2402 ± 0.0001(+0.0065)
	<i>Loss_{Eu+Eu}</i>	0.7772 ± 0.0001(-0.0093)	1.2266 ± 0.0001(-0.0071)
	<i>Loss_{Eu+MSE}</i>	0.6213 ± 0.0016(-0.1652)	1.0923 ± 0.0015(-0.1414)
PPDB _{syn+ant+}	Original	0.4734	0.0355
	Lin[4](<i>Loss_{Cos+MSE}</i>)	0.4853 ± 0.0001(+0.0119)	0.0435 ± 0.0001(+0.0119)
	<i>Loss_{Cos+Eu}</i>	0.4755 ± 0.0001(+0.0021)	0.0377 ± 0.0001(+0.0022)
	<i>Loss_{Eu+Eu}</i>	0.4754 ± 0.0001(+0.0020)	0.0371 ± 0.0001(+0.0016)
	<i>Loss_{Eu+MSE}</i>	0.4981 ± 0.0008(+0.0247)	0.0558 ± 0.0008(+0.0203)

表 4.9: 使用不同 PPDB 資料集調整預訓練詞向量 GloVe_{42B}。數據標示部分，粗體為最佳數據。

	Loss Function	$\text{loss}_{\text{Cosine}}$ (vs. Original)	$\text{loss}_{\text{Euclidean}}$ (vs. Original)
PPDB _{syn}	Original	0.6724	1.1476
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.5096 ± 0.0004 (-0.1628)	0.9953 ± 0.0003 (-0.1523)
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.6871 ± 0.0001 (+0.0147)	1.1611 ± 0.0000 (+0.0135)
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.6485 ± 0.0002 (-0.0239)	1.1272 ± 0.0003 (-0.0204)
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.3760 ± 0.0024 (-0.2964)	0.8520 ± 0.0030 (-0.2956)
PPDB _{syn+ant}	Original	0.6711	1.1369
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.5095 ± 0.0007 (-0.1616)	0.9859 ± 0.0003 (-0.1510)
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.6843 ± 0.0004 (+0.0132)	1.1490 ± 0.0002 (+0.0060)
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.6469 ± 0.0009 (-0.0242)	1.1163 ± 0.0008 (-0.0206)
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.3758 ± 0.0033 (-0.2953)	0.8420 ± 0.0040 (-0.2949)
PPDB _{syn+ant+}	Original	0.4817	0.0403
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.4957 ± 0.0005 (+0.0140)	0.0512 ± 0.0004 (+0.0109)
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.4844 ± 0.0001 (+0.0027)	0.0430 ± 0.0001 (+0.0027)
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.4844 ± 0.0002 (+0.0027)	0.0423 ± 0.0001 (+0.0020)
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.4949 ± 0.0007 (+0.0132)	0.0504 ± 0.0007 (+0.0101)

表 4.10: 使用不同 PPDB 資料集調整預訓練詞向量 Word2Vec。數據標示部分，粗體為最佳數據。

	Loss Function	$\text{loss}_{\text{Cosine}}$ (vs. Original)	$\text{loss}_{\text{Euclidean}}$ (vs. Original)
PPDB _{syn}	Original	0.6710	1.1488
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.3808 ± 0.0003 (-0.2902)	0.8627 ± 0.0003 (-0.2801)
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.6979 ± 0.0001 (+0.0269)	1.1727 ± 0.0001 (+0.0239)
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.6397 ± 0.0006 (-0.0313)	1.1220 ± 0.0005 (-0.0268)
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.3237 ± 0.0046 (-0.3473)	0.7849 ± 0.0060 (-0.3639)
PPDB _{syn+ant}	Original	0.6694	1.1378
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.3819 ± 0.0004 (-0.2875)	0.8544 ± 0.0005 (-0.2834)
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.6961 ± 0.0002 (+0.0267)	1.1614 ± 0.0001 (+0.0236)
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.6376 ± 0.0007 (-0.0318)	1.1106 ± 0.0006 (-0.0272)
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.3269 ± 0.0055 (-0.3425)	0.7792 ± 0.0073 (-0.3586)
PPDB _{syn+ant+}	Original	0.4719	0.0312
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.4892 ± 0.0001 (+0.0173)	0.0430 ± 0.0001 (+0.0118)
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.4740 ± 0.0000 (+0.0021)	0.0338 ± 0.0000 (+0.0026)
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.4775 ± 0.0001 (+0.0066)	0.0356 ± 0.0001 (+0.0044)
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.5080 ± 0.0010 (+0.0361)	0.0697 ± 0.0013 (+0.0385)

表 4.11: 使用不同 PPDB 資料集調整預訓練詞向量 fastText。數據標示部分，粗體為最佳數據。

	Loss Function	$\text{loss}_{\text{Cosine}}$ (vs. Original)	$\text{loss}_{\text{Euclidean}}$ (vs. Original)
PPDB _{syn}	Original	0.5529	1.0403
	Lin[4]($\text{Loss}_{\text{Cos}+MSE}$)	0.0701 ± 0.0002 (-0.4828)	0.3473 ± 0.0006 (-0.6930)
	$\text{Loss}_{\text{Cos}+Eu}$	0.5842 ± 0.0002 (+0.0313)	1.0706 ± 0.0002 (+0.0303)
	Loss_{Eu+Eu}	0.5934 ± 0.0004 (+0.0405)	1.0792 ± 0.0004 (+0.0389)
	Loss_{Eu+MSE}	0.4398 ± 0.0255 (-0.1131)	0.9211 ± 0.0283 (-0.1192)
PPDB _{syn+ant}	Original	0.5526	1.0306
	Lin[4]($\text{Loss}_{\text{Cos}+MSE}$)	0.0740 ± 0.0001 (-0.4786)	0.3435 ± 0.0003 (-0.6871)
	$\text{Loss}_{\text{Cos}+Eu}$	0.5839 ± 0.0001 (+0.0313)	1.0608 ± 0.0001 (+0.0302)
	Loss_{Eu+Eu}	0.5935 ± 0.0008 (+0.0409)	1.0694 ± 0.0015 (+0.0388)
	Loss_{Eu+MSE}	0.4283 ± 0.0260 (-0.1243)	0.8989 ± 0.0289 (-0.1317)
PPDB _{syn+ant+}	Original	0.4588	0.0164
	Lin[4]($\text{Loss}_{\text{Cos}+MSE}$)	0.4733 ± 0.0000 (+0.0415)	0.0132 ± 0.0001 (-0.0032)
	$\text{Loss}_{\text{Cos}+Eu}$	0.4614 ± 0.0002 (+0.0026)	0.0200 ± 0.0001 (+0.0036)
	Loss_{Eu+Eu}	0.4621 ± 0.0002 (+0.0033)	0.0210 ± 0.0001 (+0.0046)
	Loss_{Eu+MSE}	0.4776 ± 0.0008 (+0.0188)	0.0304 ± 0.0012 (+0.0140)

表 4.12: 使用不同 Lexicon 資料集調整預訓練詞向量 GloVe_{6B}。數據標示部分，粗體為最佳數據。

	Loss Function	$\text{loss}_{\text{Cosine}}$ (vs. Original)	$\text{loss}_{\text{Euclidean}}$ (vs. Original)
Lexicon _{syn}	Original	0.9089	1.3434
	Lin[4]($\text{Loss}_{\text{Cos}+MSE}$)	0.7547 ± 0.0003 (-0.1542)	1.2212 ± 0.0002 (-0.1222)
	$\text{Loss}_{\text{Cos}+Eu}$	0.9118 ± 0.0001 (+0.0029)	1.3458 ± 0.0000 (+0.0024)
	Loss_{Eu+Eu}	0.8981 ± 0.0001 (-0.0108)	1.3354 ± 0.0001 (-0.0080)
	Loss_{Eu+MSE}	0.7319 ± 0.0028 (-0.1770)	1.2021 ± 0.0025 (-0.1413)
Lexicon _{syn+ant}	Original	0.8748	1.2339
	Lin[4]($\text{Loss}_{\text{Cos}+MSE}$)	0.7331 ± 0.0003 (-0.1417)	1.1216 ± 0.0002 (-0.1123)
	$\text{Loss}_{\text{Cos}+Eu}$	0.8773 ± 0.0004 (+0.0025)	1.2362 ± 0.0000 (+0.0023)
	Loss_{Eu+Eu}	0.8648 ± 0.0001 (-0.0100)	1.2265 ± 0.0000 (-0.0074)
	Loss_{Eu+MSE}	0.7107 ± 0.0047 (-0.1641)	1.1026 ± 0.0041 (-0.1313)
Lexicon _{syn+ant+}	Original	0.7417	0.7964
	Lin[4]($\text{Loss}_{\text{Cos}+MSE}$)	0.6567 ± 0.0004 (-0.0850)	0.7296 ± 0.0004 (-0.0668)
	$\text{Loss}_{\text{Cos}+Eu}$	0.7434 ± 0.0002 (+0.0017)	0.7979 ± 0.0001 (+0.0015)
	Loss_{Eu+Eu}	0.7360 ± 0.0002 (-0.0057)	0.7922 ± 0.0001 (-0.0042)
	Loss_{Eu+MSE}	0.6437 ± 0.0010 (-0.0980)	0.7343 ± 0.0346 (-0.0621)

表 4.13: 使用不同 Lexicon 資料集調整預訓練詞向量 GloVe_{42B}。數據標示部分，粗體為最佳數據。

	Loss Function	$\text{loss}_{\text{Cosine}}$ (vs. Original)	$\text{loss}_{\text{Euclidean}}$ (vs. Original)
Lexicon _{syn}	Original	0.7972	1.2561
	Lin[4]($\text{Loss}_{\text{Cos}+MSE}$)	0.5929 ± 0.0007 (-0.2043)	1.0793 ± 0.0007 (-0.1768)
	$\text{Loss}_{\text{Cos}+Eu}$	0.8048 ± 0.0004 (+0.0076)	1.2625 ± 0.0001 (+0.0064)
	Loss_{Eu+Eu}	0.7638 ± 0.0003 (-0.0334)	1.2294 ± 0.0002 (-0.0267)
	Loss_{Eu+MSE}	0.4431 ± 0.0019 (-0.3541)	0.9275 ± 0.0021 (-0.3286)
Lexicon _{syn+ant}	Original	0.7722	1.1537
	Lin[4]($\text{Loss}_{\text{Cos}+MSE}$)	0.5846 ± 0.0005 (-0.1876)	0.9911 ± 0.0004 (-0.1626)
	$\text{Loss}_{\text{Cos}+Eu}$	0.7797 ± 0.0002 (+0.0075)	1.1600 ± 0.0001 (+0.0063)
	Loss_{Eu+Eu}	0.7414 ± 0.0003 (-0.0308)	1.1291 ± 0.0003 (-0.0246)
	Loss_{Eu+MSE}	0.4473 ± 0.0049 (-0.3249)	0.8519 ± 0.0055 (-0.3018)
Lexicon _{syn+ant+}	Original	0.6807	0.7490
	Lin[4]($\text{Loss}_{\text{Cos}+MSE}$)	0.5651 ± 0.0002 (-0.1156)	0.6502 ± 0.0002 (-0.0988)
	$\text{Loss}_{\text{Cos}+Eu}$	0.6852 ± 0.0003 (+0.0045)	0.7528 ± 0.0001 (+0.0038)
	Loss_{Eu+Eu}	0.6613 ± 0.0001 (-0.0194)	0.7336 ± 0.0001 (-0.0154)
	Loss_{Eu+MSE}	0.4756 ± 0.0029 (-0.2051)	0.5606 ± 0.0031 (-0.1884)

表 4.14: 使用不同 Lexicon 資料集調整預訓練詞向量 Word2Vec。數據標示部分，粗體為最佳數據。

	Loss Function	$\text{loss}_{\text{Cosine}}$ (vs. Original)	$\text{loss}_{\text{Euclidean}}$ (vs. Original)
Lexicon _{syn}	Original	0.7694	1.2366
	Lin[4]($\text{Loss}_{\text{Cos}+MSE}$)	0.4369 ± 0.0004 (-0.3325)	0.9275 ± 0.0004 (-0.3091)
	$\text{Loss}_{\text{Cos}+Eu}$	0.7863 ± 0.0012 (+0.0169)	1.2504 ± 0.0009 (+0.0138)
	Loss_{Eu+Eu}	0.7274 ± 0.0003 (-0.0420)	1.2023 ± 0.0002 (-0.0343)
	Loss_{Eu+MSE}	0.3682 ± 0.0049 (-0.4012)	0.8391 ± 0.0062 (-0.3975)
Lexicon _{syn+ant}	Original	0.7470	1.1360
	Lin[4]($\text{Loss}_{\text{Cos}+MSE}$)	0.4416 ± 0.0002 (-0.3054)	0.8521 ± 0.0002 (-0.2839)
	$\text{Loss}_{\text{Cos}+Eu}$	0.7621 ± 0.0012 (+0.0151)	1.1484 ± 0.0009 (+0.0124)
	Loss_{Eu+Eu}	0.7084 ± 0.0003 (-0.0386)	1.1045 ± 0.0003 (-0.0315)
	Loss_{Eu+MSE}	0.3809 ± 0.0064 (-0.3661)	0.7739 ± 0.0080 (-0.3621)
Lexicon _{syn+ant+}	Original	0.6593	0.7330
	Lin[4]($\text{Loss}_{\text{Cos}+MSE}$)	0.4695 ± 0.0000 (-0.1898)	0.5580 ± 0.0001 (-0.1750)
	$\text{Loss}_{\text{Cos}+Eu}$	0.6682 ± 0.0003 (+0.0089)	0.7404 ± 0.0003 (+0.0074)
	Loss_{Eu+Eu}	0.6368 ± 0.0004 (-0.0225)	0.7148 ± 0.0003 (-0.0182)
	Loss_{Eu+MSE}	0.4327 ± 0.0051 (-0.2266)	0.5120 ± 0.0061 (-0.2210)

表 4.15: 使用不同 Lexicon 資料集調整預訓練詞向量 fastText。數據標示部分，粗體為最佳數據。

	Loss Function	$\text{loss}_{\text{Cosine}}$ (vs. Original)	$\text{loss}_{\text{Euclidean}}$ (vs. Original)
Lexicon _{syn}	Original	0.6542	1.1386
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.1125 ± 0.0002 (-0.5417)	0.4254 ± 0.0005 (-0.7132)
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.6858 ± 0.0002 (+0.0316)	1.1663 ± 0.0002 (+0.0277)
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.6967 ± 0.0004 (+0.0425)	1.1756 ± 0.0003 (+0.0370)
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.4918 ± 0.0115 (-0.1624)	0.9785 ± 0.0123 (-0.1601)
Lexicon _{syn+ant}	Original	0.6411	1.0459
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.1420 ± 0.0001 (-0.4991)	0.3877 ± 0.0004 (-0.6582)
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.6702 ± 0.0003 (+0.0291)	1.0714 ± 0.0003 (+0.0255)
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.6804 ± 0.0008 (+0.0393)	1.0801 ± 0.0006 (+0.0342)
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.4894 ± 0.0087 (-0.1517)	0.8962 ± 0.0092 (-0.1497)
Lexicon _{syn+ant+}	Original	0.5888	0.6730
	Lin[4]($\text{Loss}_{\text{Cos}+\text{MSE}}$)	0.2709 ± 0.0001 (-0.3179)	0.2540 ± 0.0001 (-0.4190)
	$\text{Loss}_{\text{Cos}+\text{Eu}}$	0.6085 ± 0.0001 (+0.0197)	0.6903 ± 0.0001 (+0.0173)
	$\text{Loss}_{\text{Eu}+\text{Eu}}$	0.6152 ± 0.0003 (+0.0264)	0.6960 ± 0.0002 (+0.0230)
	$\text{Loss}_{\text{Eu}+\text{MSE}}$	0.4851 ± 0.0062 (-0.1037)	0.5709 ± 0.0064 (-0.1021)

4.1.3.4 相似度任務

本節實驗以相似度實驗做為評估方式，比較調整前後的詞向量對相似度分數資料集的評估結果，結果如表 4.16所示。我們發現若以相似度任務做為模型表現的評估，其結果顯示調整後的詞向量較調整前差。

表 4.16: 調整前後的詞向量對相似度任務 (SimLex-999) 的實驗結果，表格中數據為斯皮爾曼等級相關係數。數據標示部分，粗體為最佳數據。

	Loss Function	GloVe _{6B}	GloVe _{42B}	Word2Vec	fastText
	Original	0.3692	0.3725	0.3579	0.4430+
FrameNet _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.3327 ± 0.0020	0.3190 ± 0.0045	0.2995 ± 0.0007	0.1236 ± 0.0032
	$Loss_{Cos+Eu}$	0.2414 ± 0.0011	0.3509 ± 0.0041	0.3550 ± 0.0016	0.3229 ± 0.0030
	$Loss_{Eu+Eu}$	0.3123 ± 0.0019	0.3766 ± 0.0022	0.3443 ± 0.0013	0.1158 ± 0.0025
	$Loss_{Eu+MSE}$	0.3028 ± 0.0044	0.2929 ± 0.0042	0.1571 ± 0.0020	0.1505 ± 0.0119
PPDB _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.3320 ± 0.0045	0.3334 ± 0.0116	0.2820 ± 0.0020	0.2536 ± 0.0063
	$Loss_{Cos+Eu}$	0.3641 ± 0.0017	0.3651 ± 0.0088	0.3533 ± 0.0017	0.4328 ± 0.0031
	$Loss_{Eu+Eu}$	0.3670 ± 0.0019	0.3707 ± 0.0018	0.3475 ± 0.0022	0.4293 ± 0.0019
	$Loss_{Eu+MSE}$	0.2872 ± 0.0159	0.3124 ± 0.0061	0.1229 ± 0.0108	0.2227 ± 0.0216
PPDB _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.3298 ± 0.0035	0.3229 ± 0.0056	0.2791 ± 0.0042	0.2553 ± 0.0063
	$Loss_{Cos+Eu}$	0.3632 ± 0.0025	0.3652 ± 0.0066	0.3536 ± 0.0047	0.4310 ± 0.0029
	$Loss_{Eu+Eu}$	0.3681 ± 0.0015	0.3710 ± 0.0014	0.3457 ± 0.0030	0.4275 ± 0.0029
	$Loss_{Eu+MSE}$	0.2972 ± 0.0089	0.3191 ± 0.0156	0.1247 ± 0.0143	0.2273 ± 0.0080
PPDB _{syn+ant+}	Lin[4]($Loss_{Cos+MSE}$)	0.3350 ± 0.0041	0.3287 ± 0.0043	0.2918 ± 0.0030	0.2706 ± 0.0057
	$Loss_{Cos+Eu}$	0.3594 ± 0.0024	0.3577 ± 0.0058	0.3463 ± 0.0023	0.4271 ± 0.0044
	$Loss_{Eu+Eu}$	0.3651 ± 0.0017	0.3673 ± 0.0042	0.3414 ± 0.0018	0.4221 ± 0.0020
	$Loss_{Eu+MSE}$	0.2734 ± 0.0141	0.2954 ± 0.0142	0.1011 ± 0.0110	0.2594 ± 0.0116
WordNet _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.3360 ± 0.0011	0.3397 ± 0.0025	0.2974 ± 0.0008	0.2421 ± 0.0019
	$Loss_{Cos+Eu}$	0.2483 ± 0.0014	0.3598 ± 0.0028	0.3516 ± 0.0014	0.3065 ± 0.0007
	$Loss_{Eu+Eu}$	0.2370 ± 0.0016	0.3754 ± 0.0026	0.3431 ± 0.0008	0.4281 ± 0.0019
	$Loss_{Eu+MSE}$	0.2952 ± 0.0052	0.3208 ± 0.0071	0.1649 ± 0.0025	0.2258 ± 0.0036
WordNet _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.3338 ± 0.0019	0.3153 ± 0.0034	0.2494 ± 0.0011	0.2459 ± 0.0040
	$Loss_{Cos+Eu}$	0.2494 ± 0.0011	0.3592 ± 0.0017	0.3502 ± 0.0005	0.3060 ± 0.0003
	$Loss_{Eu+Eu}$	0.2293 ± 0.0022	0.3679 ± 0.0022	0.3451 ± 0.0010	0.4294 ± 0.0016
	$Loss_{Eu+MSE}$	0.2917 ± 0.0038	0.3282 ± 0.0080	0.1701 ± 0.0036	0.2259 ± 0.0063
Lexicon _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.3127 ± 0.0036	0.3042 ± 0.0061	0.2528 ± 0.0048	0.1855 ± 0.0027
	$Loss_{Cos+Eu}$	0.3563 ± 0.0019	0.3527 ± 0.0023	0.3420 ± 0.0028	0.4193 ± 0.0050
	$Loss_{Eu+Eu}$	0.3560 ± 0.0027	0.3522 ± 0.0022	0.3358 ± 0.0017	0.4161 ± 0.0016
	$Loss_{Eu+MSE}$	0.2134 ± 0.0032	0.2286 ± 0.0030	0.0770 ± 0.0127	0.1879 ± 0.0097
Lexicon _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.3109 ± 0.0047	0.3047 ± 0.0040	0.2516 ± 0.0018	0.1859 ± 0.0053
	$Loss_{Cos+Eu}$	0.3565 ± 0.0017	0.3560 ± 0.0026	0.3436 ± 0.0022	0.4169 ± 0.0036
	$Loss_{Eu+Eu}$	0.3569 ± 0.0018	0.3557 ± 0.0054	0.3360 ± 0.0025	0.4161 ± 0.0044
	$Loss_{Eu+MSE}$	0.2064 ± 0.0111	0.2391 ± 0.0204	0.0630 ± 0.0099	0.1786 ± 0.0166
Lexicon _{syn+ant+}	Lin[4]($Loss_{Cos+MSE}$)	0.3134 ± 0.0048	0.3112 ± 0.0023	0.2644 ± 0.0005	0.1978 ± 0.0049
	$Loss_{Cos+Eu}$	0.3566 ± 0.0012	0.3523 ± 0.0042	0.3444 ± 0.0030	0.4225 ± 0.0011
	$Loss_{Eu+Eu}$	0.3572 ± 0.0023	0.3529 ± 0.0099	0.3338 ± 0.0025	0.4166 ± 0.0015
	$Loss_{Eu+MSE}$	0.2081 ± 0.0045	0.2438 ± 0.0115	0.0725 ± 0.0109	0.1848 ± 0.0119

4.1.3.5 調整結果展示

本章節展示調整前後之詞向量以視覺化呈現的結果。本研究將詞向量降維並以二維向量空間呈現。如圖 4.1、4.2所示，我們在向量空間中呈現出與“his”最相近的一百個字，並以表 4.17以相似度呈現其最相近的五個字。

表 4.17: 調整結果展示

	Top-5 similar words of “his”				
GloVe _{42B}	her (0.8270)	he (0.8268)	him (0.8192)	himself (0.7715)	my (0.7569)
Our Work	he (0.8211)	him (0.8133)	her (0.7960)	their (0.7630)	own (0.7484)

四、實驗結果與分析

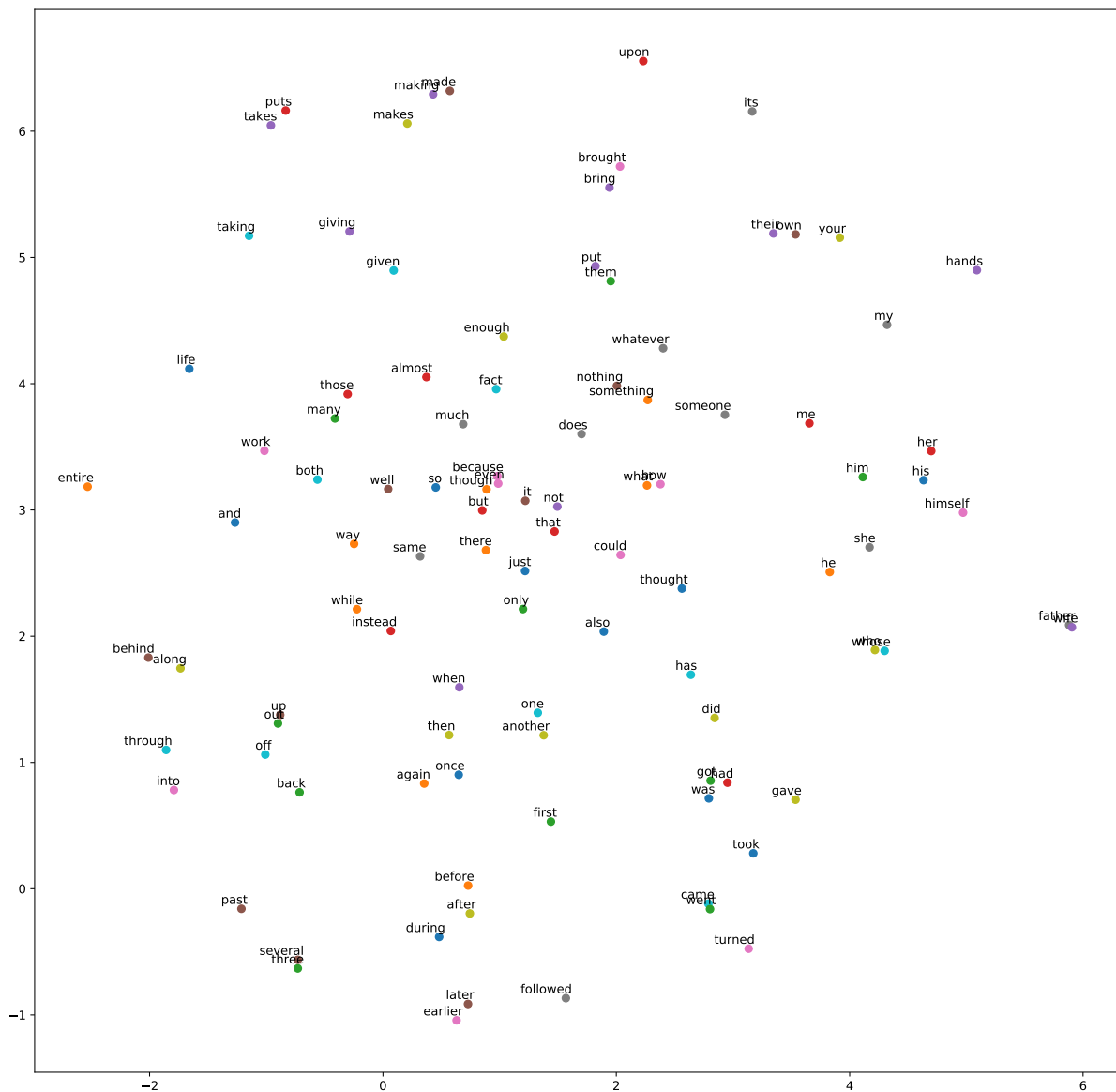


圖 4.2: Top-100 similar words of “his” in our work.

4.2 Task 2：自然語言處理任務

從上階段實驗,我們發現調整後表現最好的詞向量為 [4]($Loss_{Cos+MSE}$) 及使用 $Loss_{Eu+MSE}$ 調整的詞向量,兩者互有優劣,其次依序是使用 $Loss_{Eu+Eu}$ 調整的詞向量、調整前的詞向量、使用 $Loss_{Cos+Eu}$ 調整的詞向量。接著,我們使用前四者進行下游實驗。本研究以情感分析類的自然語言處理任務做為下游任務,此種任務目標是要判斷一段敘述屬於正面 (positive) 還是負面 (negative)。我們認為我們以同義詞反義詞資料集調整後的詞向量,應能較調整前的詞向量更精準地預測出任務中評論何者為正面的、何者為負面的。

4.2.1 模型設置

為了簡單化模型,使輸出結果受詞向量影響較大,而非藉由模型的複雜度來提升訓練結果,我們使用基本的長短期記憶模型,且只設置一層 LSTM 來進行訓練,並以 Adam Optimizer[35] 做為模型的優化器。

4.2.2 IMDb

4.2.2.1 任務介紹

IMDb 為一網路電影資料庫,其網站上有極為豐富的電影作品資訊,包含超過 650 萬部作品、超過 1 億的人物資訊及 600 萬註冊會員所留下的電影評論。我們所使用的資料集⁴包含 5 萬筆做為情感分析的電影評論,其中 25,000 筆備標註為 positive,另外 25,000 筆則為 negative。

⁴<https://ai.stanford.edu/amaas/data/sentiment/>

4.2.2.2 實驗結果

根據圖 4.3及 4.18表~ 4.20我們可以發現，不管使用何種同反義詞資料集及損失函數對 GloVe_{6B}、GloVe_{42B}、Word2Vec 進行訓練，其訓練後的詞向量在 IMDB 任務的表現均較調整前來的差。且儘管使用 [4] 所提出的損失函數 $Loss_{Cos+MSE}$ 及 $Loss_{Eu+MSE}$ 來調整的詞向量在第一階段實驗結果最佳，但其在下游任務的表現反而都較以 $Loss_{Eu+Eu}$ 調整的詞向量差，僅有以同反義詞資料集 WordNet_{syn}、WordNet_{syn+ant} 及損失函數 $Loss_{Eu+MSE}$ 調整的詞向量稍優於 $Loss_{Eu+Eu}$ 調整的詞向量。

而根據表 4.21的結果顯示，使用損失函數 $Loss_{Eu+MSE}$ 及 $Loss_{Eu+Eu}$ 調整的詞向量有帶來比調整前的詞向量好的結果，但於上階段實驗表現較好的 [4] 所提出的損失函數 $Loss_{Cos+MSE}$ 調整的詞向量仍然無法超過調整前的詞向量，且為最差。

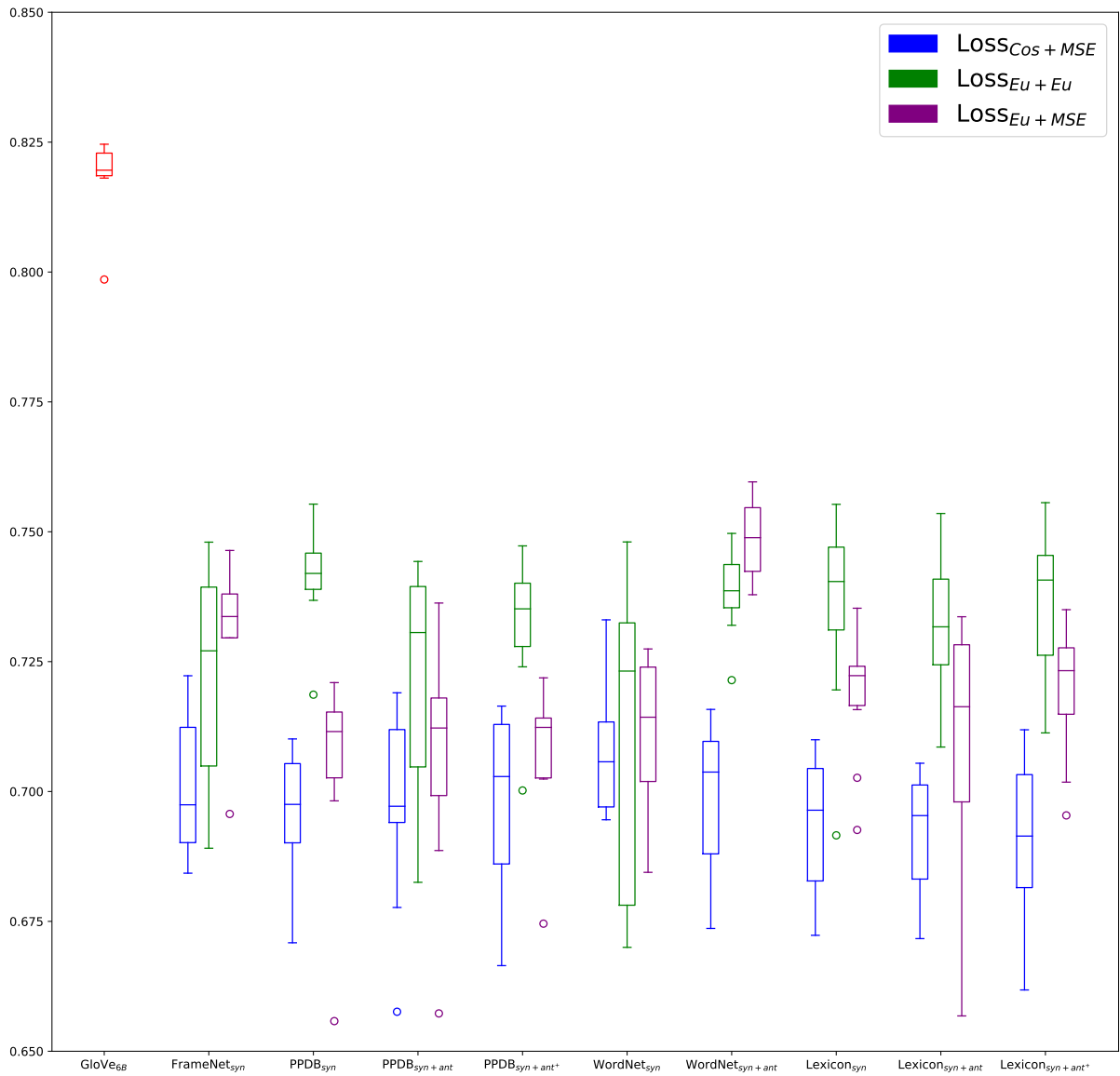


圖 4.3: 詞向量 GloVe_{6B} 及以 GloVe_{6B} 調整的詞向量 IMDb 預測之結果。此圖調整後之詞向量未經 finetune。

表 4.18: 詞向量 GloVe_{6B} 及以 GloVe_{6B} 調整的詞向量 IMDb 預測之結果。數據標示部分，粗體為最佳數據。

	Loss Function	Test Accuracy (詞向量未經 finetune)	Test Accuracy (詞向量進行 finetune)
GloVe _{6B}		0.8187 ± 0.0074	0.8200 ± 0.0036
FrameNet _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.7008 ± 0.0141	0.6975 ± 0.0163
	$Loss_{Eu+Eu}$	0.7227 ± 0.0218	0.7278 ± 0.0319
	$Loss_{Eu+MSE}$	0.7122 ± 0.0628	0.7275 ± 0.0088
PPDB _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.6960 ± 0.0124	0.6895 ± 0.0259
	$Loss_{Eu+Eu}$	0.7417 ± 0.0100	0.7260 ± 0.0177
	$Loss_{Eu+MSE}$	0.7051 ± 0.0187	0.7052 ± 0.0105
PPDB _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.6975 ± 0.0189	0.6901 ± 0.0205
	$Loss_{Eu+Eu}$	0.7203 ± 0.0238	0.7424 ± 0.0101
	$Loss_{Eu+MSE}$	0.7066 ± 0.0218	0.7108 ± 0.0116
PPDB _{syn+ant+}	Lin[4]($Loss_{Cos+MSE}$)	0.6972 ± 0.0184	0.7048 ± 0.0240
	$Loss_{Eu+Eu}$	0.7320 ± 0.0133	0.7354 ± 0.0128
	$Loss_{Eu+MSE}$	0.6928 ± 0.0495	0.6784 ± 0.0520
WordNet _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.7014 ± 0.0262	0.7059 ± 0.0084
	$Loss_{Eu+Eu}$	0.7108 ± 0.0311	0.7092 ± 0.0094
	$Loss_{Eu+MSE}$	0.7117 ± 0.0142	0.7200 ± 0.0241
WordNet _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.6945 ± 0.0234	0.7069 ± 0.0073
	$Loss_{Eu+Eu}$	0.7387 ± 0.0083	0.7240 ± 0.0209
	$Loss_{Eu+MSE}$	0.7484 ± 0.0075	0.7428 ± 0.0165
Lexicon _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.6892 ± 0.0218	0.7023 ± 0.0127
	$Loss_{Eu+Eu}$	0.7356 ± 0.0187	0.7280 ± 0.0185
	$Loss_{Eu+MSE}$	0.7192 ± 0.0132	0.7205 ± 0.0068
Lexicon _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.6918 ± 0.0114	0.6808 ± 0.0247
	$Loss_{Eu+Eu}$	0.7329 ± 0.0136	0.7317 ± 0.0117
	$Loss_{Eu+MSE}$	0.7100 ± 0.0233	0.7190 ± 0.0187
Lexicon _{syn+ant+}	Lin[4]($Loss_{Cos+MSE}$)	0.6899 ± 0.0164	0.6973 ± 0.0142
	$Loss_{Eu+Eu}$	0.7361 ± 0.0138	0.7319 ± 0.0121
	$Loss_{Eu+MSE}$	0.7194 ± 0.0127	0.6920 ± 0.0347

表 4.19: 詞向量 GloVe_{42B} 及以 GloVe_{42B} 調整的詞向量 IMDb 預測之結果。數據標示部分，粗體為最佳數據。

	Loss Function	Test Accuracy (詞向量未經 finetune)	Test Accuracy (詞向量進行 finetune)
GloVe _{42B}		0.8225 ± 0.0066	0.8127 ± 0.0147
FrameNet _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.7076 ± 0.0146	0.6985 ± 0.0226
	$Loss_{Eu+Eu}$	0.7293 ± 0.0112	0.7254 ± 0.0076
	$Loss_{Eu+MSE}$	0.7176 ± 0.0126	0.7208 ± 0.0137
PPDB _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.6935 ± 0.0160	0.7038 ± 0.0128
	$Loss_{Eu+Eu}$	0.7256 ± 0.0118	0.7278 ± 0.0110
	$Loss_{Eu+MSE}$	0.7033 ± 0.0090	0.7092 ± 0.0127
PPDB _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.7057 ± 0.0092	0.7019 ± 0.0088
	$Loss_{Eu+Eu}$	0.7238 ± 0.0105	0.7221 ± 0.0025
	$Loss_{Eu+MSE}$	0.7091 ± 0.0121	0.7102 ± 0.0098
PPDB _{syn+ant+}	Lin[4]($Loss_{Cos+MSE}$)	0.6988 ± 0.0272	0.6977 ± 0.0194
	$Loss_{Eu+Eu}$	0.7264 ± 0.0115	0.7286 ± 0.0173
	$Loss_{Eu+MSE}$	0.6939 ± 0.0111	0.6981 ± 0.0090
WordNet _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.7036 ± 0.0119	0.7035 ± 0.0112
	$Loss_{Eu+Eu}$	0.7288 ± 0.0138	0.7322 ± 0.0102
	$Loss_{Eu+MSE}$	0.7041 ± 0.0212	0.7065 ± 0.0203
WordNet _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.6854 ± 0.0369	0.7015 ± 0.0088
	$Loss_{Eu+Eu}$	0.7282 ± 0.0206	0.7314 ± 0.0103
	$Loss_{Eu+MSE}$	0.7017 ± 0.0199	0.7132 ± 0.0155
Lexicon _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.6970 ± 0.0123	0.7032 ± 0.0136
	$Loss_{Eu+Eu}$	0.7262 ± 0.0114	0.7331 ± 0.0060
	$Loss_{Eu+MSE}$	0.6993 ± 0.0227	0.7147 ± 0.0098
Lexicon _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.6998 ± 0.0161	0.6877 ± 0.0236
	$Loss_{Eu+Eu}$	0.7294 ± 0.0137	0.7281 ± 0.0089
	$Loss_{Eu+MSE}$	0.7081 ± 0.0116	0.7119 ± 0.0059
Lexicon _{syn+ant+}	Lin[4]($Loss_{Cos+MSE}$)	0.7059 ± 0.0093	0.6960 ± 0.0189
	$Loss_{Eu+Eu}$	0.7190 ± 0.0185	0.7139 ± 0.0218
	$Loss_{Eu+MSE}$	0.6996 ± 0.0152	0.6985 ± 0.0212

表 4.20: 詞向量 Word2Vec 及以 Word2Vec 調整的詞向量 IMDb 預測之結果。數據標示部分，粗體為最佳數據。

	Loss Function	Test Accuracy (詞向量未經 finetune)	Test Accuracy (詞向量進行 finetune)
Word2Vec		0.8041 ± 0.0064	0.8082 ± 0.0067
FrameNet _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.7022 ± 0.0162	0.7116 ± 0.0115
	$Loss_{Eu+Eu}$	0.7228 ± 0.0232	0.7287 ± 0.0035
	$Loss_{Eu+MSE}$	0.7229 ± 0.0040	0.7202 ± 0.0103
PPDB _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.6642 ± 0.0208	0.6532 ± 0.0397
	$Loss_{Eu+Eu}$	0.7264 ± 0.0099	0.6998 ± 0.0277
	$Loss_{Eu+MSE}$	0.6565 ± 0.0152	0.6160 ± 0.0346
PPDB _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.6576 ± 0.0172	0.6633 ± 0.0181
	$Loss_{Eu+Eu}$	0.7136 ± 0.0142	0.7199 ± 0.0233
	$Loss_{Eu+MSE}$	0.6361 ± 0.0156	0.6570 ± 0.0189
PPDB _{syn+ant+}	Lin[4]($Loss_{Cos+MSE}$)	0.6654 ± 0.0200	0.6627 ± 0.0090
	$Loss_{Eu+Eu}$	0.7198 ± 0.0124	0.7301 ± 0.0120
	$Loss_{Eu+MSE}$	0.6408 ± 0.0160	0.6335 ± 0.0252
WordNet _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.6510 ± 0.0331	0.6650 ± 0.0284
	$Loss_{Eu+Eu}$	0.7207 ± 0.0223	0.7302 ± 0.0023
	$Loss_{Eu+MSE}$	0.7036 ± 0.0236	0.7162 ± 0.0203
WordNet _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.6979 ± 0.0202	0.6978 ± 0.0125
	$Loss_{Eu+Eu}$	0.7269 ± 0.0174	0.7261 ± 0.0068
	$Loss_{Eu+MSE}$	0.7106 ± 0.0207	0.7111 ± 0.0130
Lexicon _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.6701 ± 0.0125	0.6669 ± 0.0152
	$Loss_{Eu+Eu}$	0.7174 ± 0.0151	0.7239 ± 0.0169
	$Loss_{Eu+MSE}$	0.6584 ± 0.0406	0.6722 ± 0.0158
Lexicon _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.6696 ± 0.0098	0.6643 ± 0.0112
	$Loss_{Eu+Eu}$	0.7081 ± 0.0311	0.7093 ± 0.0223
	$Loss_{Eu+MSE}$	0.6597 ± 0.0220	0.6696 ± 0.0118
Lexicon _{syn+ant+}	Lin[4]($Loss_{Cos+MSE}$)	0.6559 ± 0.0157	0.6632 ± 0.0244
	$Loss_{Eu+Eu}$	0.7169 ± 0.0171	0.7032 ± 0.0237
	$Loss_{Eu+MSE}$	0.6564 ± 0.0304	0.6595 ± 0.0278

表 4.21: 詞向量 fastText 及以 fastText 調整的詞向量 IMDb 預測之結果。數據標示部分，粗體為最佳數據。

	Loss Function	Test Accuracy (詞向量未經 finetune)	Test Accuracy (詞向量進行 finetune)
fastText		0.7095 ± 0.0151	0.7157 ± 0.0062
FrameNet _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.6653 ± 0.0134	0.6726 ± 0.0094
	$Loss_{Eu+Eu}$	0.6570 ± 0.0266	0.6743 ± 0.0178
	$Loss_{Eu+MSE}$	0.7535 ± 0.0121	0.7524 ± 0.0097
PPDB _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.6421 ± 0.0175	0.6544 ± 0.0062
	$Loss_{Eu+Eu}$	0.7424 ± 0.0129	0.7426 ± 0.0078
	$Loss_{Eu+MSE}$	0.7168 ± 0.0268	0.7337 ± 0.0131
PPDB _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.6533 ± 0.0112	0.6564 ± 0.0138
	$Loss_{Eu+Eu}$	0.7481 ± 0.0113	0.7490 ± 0.0074
	$Loss_{Eu+MSE}$	0.7244 ± 0.0306	0.7204 ± 0.0163
PPDB _{syn+ant+}	Lin[4]($Loss_{Cos+MSE}$)	0.6462 ± 0.0084	0.6377 ± 0.0116
	$Loss_{Eu+Eu}$	0.7481 ± 0.0112	0.7380 ± 0.0203
	$Loss_{Eu+MSE}$	0.7244 ± 0.0178	0.7234 ± 0.0085
WordNet _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.6349 ± 0.0261	0.6154 ± 0.0337
	$Loss_{Eu+Eu}$	0.7430 ± 0.0182	0.7419 ± 0.0154
	$Loss_{Eu+MSE}$	0.7388 ± 0.0122	0.7415 ± 0.0117
WordNet _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.6425 ± 0.0149	0.6266 ± 0.0387
	$Loss_{Eu+Eu}$	0.7446 ± 0.0147	0.7543 ± 0.0089
	$Loss_{Eu+MSE}$	0.7370 ± 0.0197	0.7430 ± 0.0037
Lexicon _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.6478 ± 0.0172	0.6323 ± 0.0260
	$Loss_{Eu+Eu}$	0.7438 ± 0.0172	0.7370 ± 0.0274
	$Loss_{Eu+MSE}$	0.7231 ± 0.0257	0.7351 ± 0.0118
Lexicon _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.6433 ± 0.0129	0.6495 ± 0.0165
	$Loss_{Eu+Eu}$	0.7505 ± 0.0080	0.7556 ± 0.0110
	$Loss_{Eu+MSE}$	0.7322 ± 0.0163	0.7223 ± 0.0190
Lexicon _{syn+ant+}	Lin[4]($Loss_{Cos+MSE}$)	0.6468 ± 0.0193	0.6437 ± 0.0249
	$Loss_{Eu+Eu}$	0.7413 ± 0.0122	0.7550 ± 0.0033
	$Loss_{Eu+MSE}$	0.7404 ± 0.0126	0.7326 ± 0.0133

4.2.3 US Airline

4.2.3.1 任務介紹

US Airline 為一推特 (Twitter) 上蒐集而來的情感分析資料集，其為使用者對於美國六家航空公司的評論。我們所使用的資料集⁵ 包含筆評論，其中 2354 筆被標註為 positive，另外 9169 筆則為 negative。

4.2.3.2 實驗結果

我們可以從實驗發現 GloVe_{6B} 及 Word2Vec 表現最好的均為調整前的詞向量，而 [4] ($Loss_{Cos+MSE}$)、 $Loss_{Eu+Eu}$ 、 $Loss_{Eu+MSE}$ 三種不同調整方法雖然在表現上互有優劣，但大部分情況下結果較好的是以損失函數 $Loss_{Eu+Eu}$ 調整的詞向量。

雖然另外兩個詞向量 GloVe_{42B} 及 fastText，表現最好的是調整後的詞向量，分別為使用 WordNet_{syn} 資料集及損失函數 $Loss_{Eu+Eu}$ 調整的 GloVe_{42B} 和使用 PPDB_{syn} 資料集及損失函數 $Loss_{Eu+Eu}$ 調整的 fastText。但其他大部分其況下調整後的詞向量均較調整前差，且使用損失函數 $Loss_{Eu+Eu}$ 調整的詞向量幾乎所有情況下都比 [4] 提出的損失函數 $Loss_{Cos+MSE}$ 及損失函數 $Loss_{Eu+MSE}$ 表現來的好。

⁵<https://www.kaggle.com/crowdflower/twitter-airline-sentiment>

表 4.22: 詞向量 GloVe_{6B} 及以 GloVe_{6B} 調整的詞向量對 US airline 預測之結果。數據標示部分，粗體為最佳數據。

	Loss Function	Test Accuracy (詞向量未經 finetune)	Test Accuracy (詞向量進行 finetune)
GloVe _{6B}		0.8252 ± 0.0353	0.8287 ± 0.0354
FrameNet _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.7956 ± 0.0000	0.7956 ± 0.0000
	$Loss_{Eu+Eu}$	0.7956 ± 0.0000	0.7956 ± 0.0000
	$Loss_{Eu+MSE}$	0.7956 ± 0.0000	0.7956 ± 0.0000
PPDB _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.7956 ± 0.0000	0.7956 ± 0.0000
	$Loss_{Eu+Eu}$	0.7977 ± 0.0064	0.7956 ± 0.0000
	$Loss_{Eu+MSE}$	0.7904 ± 0.0190	0.7956 ± 0.0000
PPDB _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.7956 ± 0.0000	0.7956 ± 0.0000
	$Loss_{Eu+Eu}$	0.8120 ± 0.0346	0.8067 ± 0.0118
	$Loss_{Eu+MSE}$	0.7964 ± 0.0026	0.7956 ± 0.0000
PPDB _{syn+ant+}	Lin[4]($Loss_{Cos+MSE}$)	0.7981 ± 0.0078	0.8042 ± 0.0000
	$Loss_{Eu+Eu}$	0.8033 ± 0.0183	0.8097 ± 0.0395
	$Loss_{Eu+MSE}$	0.7956 ± 0.0000	0.7956 ± 0.0000
WordNet _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.7956 ± 0.0000	0.8005 ± 0.0219
	$Loss_{Eu+Eu}$	0.7956 ± 0.0000	0.8022 ± 0.0292
	$Loss_{Eu+MSE}$	0.7956 ± 0.0000	0.7956 ± 0.0000
WordNet _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.8040 ± 0.0264	0.7999 ± 0.0000
	$Loss_{Eu+Eu}$	0.7956 ± 0.0000	0.7962 ± 0.0000
	$Loss_{Eu+MSE}$	0.7983 ± 0.0084	0.7974 ± 0.0079
Lexicon _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.7956 ± 0.0000	0.7956 ± 0.0000
	$Loss_{Eu+Eu}$	0.8005 ± 0.0106	0.8091 ± 0.0236
	$Loss_{Eu+MSE}$	0.7956 ± 0.0000	0.7956 ± 0.0000
Lexicon _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.7966 ± 0.0031	0.7961 ± 0.0000
	$Loss_{Eu+Eu}$	0.7966 ± 0.0031	0.8061 ± 0.0000
	$Loss_{Eu+MSE}$	0.7956 ± 0.0000	0.7956 ± 0.0000
Lexicon _{syn+ant+}	Lin[4]($Loss_{Cos+MSE}$)	0.7956 ± 0.0000	0.7956 ± 0.0000
	$Loss_{Eu+Eu}$	0.8138 ± 0.0310	0.8024 ± 0.0207
	$Loss_{Eu+MSE}$	0.7956 ± 0.0000	0.7956 ± 0.0000

表 4.23: 詞向量 GloVe_{42B} 及以 GloVe_{42B} 調整的詞向量對 US airline 預測之結果。數據標示部分，粗體為最佳數據。

	Loss Function	Test Accuracy (詞向量未經 finetune)	Test Accuracy (詞向量進行 finetune)
GloVe _{42B}		0.8138 ± 0.0310	0.8166 ± 0.0388
FrameNet _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.7995 ± 0.0120	0.7973 ± 0.0000
	$Loss_{Eu+Eu}$	0.8021 ± 0.0204	0.8031 ± 0.0000
	$Loss_{Eu+MSE}$	0.8019 ± 0.0211	0.8060 ± 0.0135
PPDB _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.7965 ± 0.0028	0.7956 ± 0.0000
	$Loss_{Eu+Eu}$	0.7956 ± 0.0000	0.8027 ± 0.0032
	$Loss_{Eu+MSE}$	0.7929 ± 0.0085	0.7965 ± 0.0038
PPDB _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.7956 ± 0.0000	0.7956 ± 0.0000
	$Loss_{Eu+Eu}$	0.7962 ± 0.0019	0.8163 ± 0.0258
	$Loss_{Eu+MSE}$	0.8166 ± 0.0291	0.7956 ± 0.0000
PPDB _{syn+ant+}	Lin[4]($Loss_{Cos+MSE}$)	0.7987 ± 0.0066	0.7996 ± 0.0000
	$Loss_{Eu+Eu}$	0.7956 ± 0.0000	0.7956 ± 0.0000
	$Loss_{Eu+MSE}$	0.7806 ± 0.0476	0.7956 ± 0.0000
WordNet _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.7956 ± 0.0000	0.8051 ± 0.0356
	$Loss_{Eu+Eu}$	0.8206 ± 0.0351	0.7910 ± 0.0191
	$Loss_{Eu+MSE}$	0.7956 ± 0.0000	0.7971 ± 0.0034
WordNet _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.8014 ± 0.0182	0.7956 ± 0.0000
	$Loss_{Eu+Eu}$	0.8039 ± 0.0184	0.7962 ± 0.0025
	$Loss_{Eu+MSE}$	0.7956 ± 0.0000	0.8008 ± 0.0000
Lexicon _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.8044 ± 0.0278	0.7984 ± 0.0000
	$Loss_{Eu+Eu}$	0.7986 ± 0.0094	0.8107 ± 0.0246
	$Loss_{Eu+MSE}$	0.7956 ± 0.0000	0.7956 ± 0.0000
Lexicon _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.7956 ± 0.0000	0.7956 ± 0.0000
	$Loss_{Eu+Eu}$	0.7976 ± 0.0063	0.7956 ± 0.0000
	$Loss_{Eu+MSE}$	0.7980 ± 0.0051	0.8014 ± 0.0259
Lexicon _{syn+ant+}	Lin[4]($Loss_{Cos+MSE}$)	0.7956 ± 0.0000	0.7987 ± 0.0139
	$Loss_{Eu+Eu}$	0.7956 ± 0.0000	0.8172 ± 0.0332
	$Loss_{Eu+MSE}$	0.7956 ± 0.0000	0.7956 ± 0.0000

表 4.24: 詞向量 Word2Vec 及以 Word2Vec 調整的詞向量對 US airline 預測之結果。數據標示部分，粗體為最佳數據。

	Loss Function	Test Accuracy (詞向量未經 finetune)	Test Accuracy (詞向量進行 finetune)
Word2Vec		0.8045 ± 0.0164	0.8006 ± 0.0188
FrameNet _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.8000 ± 0.0138	0.7965 ± 0.0000
	$Loss_{Eu+Eu}$	0.7956 ± 0.0000	0.7956 ± 0.0000
	$Loss_{Eu+MSE}$	0.7956 ± 0.0000	0.7988 ± 0.0000
PPDB _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.7965 ± 0.0028	0.7957 ± 0.0000
	$Loss_{Eu+Eu}$	0.7956 ± 0.0000	0.7956 ± 0.0000
	$Loss_{Eu+MSE}$	0.7956 ± 0.0000	0.7965 ± 0.0000
PPDB _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.7956 ± 0.0000	0.7956 ± 0.0000
	$Loss_{Eu+Eu}$	0.7962 ± 0.0019	0.7956 ± 0.0000
	$Loss_{Eu+MSE}$	0.7956 ± 0.0000	0.7956 ± 0.0000
PPDB _{syn+ant+}	Lin[4]($Loss_{Cos+MSE}$)	0.7987 ± 0.0066	0.7980 ± 0.0000
	$Loss_{Eu+Eu}$	0.7956 ± 0.0000	0.7955 ± 0.0000
	$Loss_{Eu+MSE}$	0.7956 ± 0.0000	0.7956 ± 0.0000
WordNet _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.7957 ± 0.0001	0.7963 ± 0.0000
	$Loss_{Eu+Eu}$	0.7956 ± 0.0000	0.7956 ± 0.0000
	$Loss_{Eu+MSE}$	0.8006 ± 0.0157	0.8076 ± 0.0162
WordNet _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.7983 ± 0.0085	0.7956 ± 0.0000
	$Loss_{Eu+Eu}$	0.7956 ± 0.0000	0.7956 ± 0.0000
	$Loss_{Eu+MSE}$	0.7956 ± 0.0000	0.7978 ± 0.0000
Lexicon _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.8044 ± 0.0278	0.7956 ± 0.0000
	$Loss_{Eu+Eu}$	0.7986 ± 0.0094	0.7956 ± 0.0000
	$Loss_{Eu+MSE}$	0.7956 ± 0.0000	0.7956 ± 0.0000
Lexicon _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.7956 ± 0.0000	0.7956 ± 0.0000
	$Loss_{Eu+Eu}$	0.7976 ± 0.0063	0.7982 ± 0.0083
	$Loss_{Eu+MSE}$	0.7956 ± 0.0000	0.7939 ± 0.0075
Lexicon _{syn+ant+}	Lin[4]($Loss_{Cos+MSE}$)	0.7956 ± 0.0000	0.7968 ± 0.0000
	$Loss_{Eu+Eu}$	0.7956 ± 0.0000	0.7960 ± 0.0000
	$Loss_{Eu+MSE}$	0.7962 ± 0.0019	0.7956 ± 0.0000

表 4.25: 詞向量 fastText 及以 fastText 調整的詞向量對 US airline 預測之結果。數據標示部分，粗體為最佳數據，方框為表現較調整前的詞向量好的數據。

	Loss Function	Test Accuracy (詞向量未經 finetune)	Test Accuracy (詞向量進行 finetune)
fastText		0.7956 ± 0.0000	0.7956 ± 0.0000
FrameNet _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.7969 ± 0.0039	0.7956 ± 0.0000
	$Loss_{Eu+Eu}$	0.7956 ± 0.0000	0.7985 ± 0.0090
	$Loss_{Eu+MSE}$	0.7959 ± 0.0008	0.7956 ± 0.0000
PPDB _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.7956 ± 0.0000	0.7956 ± 0.0000
	$Loss_{Eu+Eu}$	0.8073 ± 0.0231	0.7987 ± 0.0097
	$Loss_{Eu+MSE}$	0.7956 ± 0.0000	0.7956 ± 0.0000
PPDB _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.7956 ± 0.0000	0.7956 ± 0.0000
	$Loss_{Eu+Eu}$	0.7956 ± 0.0000	0.8022 ± 0.0153
	$Loss_{Eu+MSE}$	0.7956 ± 0.0000	0.7976 ± 0.0063
PPDB _{syn+ant+}	Lin[4]($Loss_{Cos+MSE}$)	0.7956 ± 0.0000	0.7956 ± 0.0000
	$Loss_{Eu+Eu}$	0.7995 ± 0.0123	0.7956 ± 0.0000
	$Loss_{Eu+MSE}$	0.7966 ± 0.0030	0.7979 ± 0.0072
WordNet _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.7956 ± 0.0000	0.7956 ± 0.0000
	$Loss_{Eu+Eu}$	0.7962 ± 0.0017	0.7956 ± 0.0000
	$Loss_{Eu+MSE}$	0.7956 ± 0.0001	0.7956 ± 0.0000
WordNet _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.7957 ± 0.0001	0.7956 ± 0.0000
	$Loss_{Eu+Eu}$	0.7956 ± 0.0000	0.7967 ± 0.0032
	$Loss_{Eu+MSE}$	0.7995 ± 0.0084	0.7956 ± 0.0000
Lexicon _{syn}	Lin[4]($Loss_{Cos+MSE}$)	0.7956 ± 0.0000	0.7956 ± 0.0000
	$Loss_{Eu+Eu}$	0.7955 ± 0.0002	0.7956 ± 0.0000
	$Loss_{Eu+MSE}$	0.7956 ± 0.0000	0.7956 ± 0.0000
Lexicon _{syn+ant}	Lin[4]($Loss_{Cos+MSE}$)	0.7956 ± 0.0000	0.7956 ± 0.0000
	$Loss_{Eu+Eu}$	0.7956 ± 0.0000	0.7966 ± 0.0030
	$Loss_{Eu+MSE}$	0.7956 ± 0.0000	0.7956 ± 0.0000
Lexicon _{syn+ant+}	Lin[4]($Loss_{Cos+MSE}$)	0.7935 ± 0.0067	0.7963 ± 0.0021
	$Loss_{Eu+Eu}$	0.8033 ± 0.0196	0.8024 ± 0.0214
	$Loss_{Eu+MSE}$	0.7956 ± 0.0000	0.7956 ± 0.0000

五、 總結

5.1 結論

根據上一章節所進行的實驗結果顯示，即使調整後獲得了更了解自然語言的字詞關係的詞向量，其在下游的自然語言處理任務無法獲得較好的結果。

從第一階段實驗，我們發現同義詞相較之下，更能讓詞向量獲得更佳的字詞資訊，且較多的字詞關係資訊也可使詞向量出現在向量上更適合的位置。另外，從實驗結果來看，表現最好的詞向量為 [4] 使用餘弦距離加上均方差調整的詞向量及使用 $Loss_{Eu+MSE}$ 調整的詞向量，其次依序是使用 $Loss_{Eu+Eu}$ 調整的詞向量、調整前的詞向量、使用 $Loss_{Cos+Eu}$ 調整的詞向量。

而在進行第二階段實驗之後，我們發現在第一階段表現最好的詞向量反而最差、第一階段表現最差的調整前的詞向量反而獲得較好的成果。因此我們認為即使讓詞向量學習到了更符合詞庫中的字詞關係的資訊，其在下游任務也不一定能獲得更好的表現。

此外，值得注意的是，與 GloVe 和 Word2Vec 只透過文本訓練的詞向量不太一樣的 fastText，其在訓練過程中加上了 sub-word 資訊進行訓練，使詞向量能夠學習到更豐富的資訊的詞向量。我們認為由於此種詞向量已經學習到了比前兩種詞向量更多不同的字詞關係，使用此預訓練詞向量訓練所得到的新的詞向量應無法得到較顯著進步的成果。但反而在進行第二階段下游任務的實驗之後發現，只有使用 fastText 訓練而得

的新的詞向量都能獲得較調整前的詞向量更好的結果。

最後，關於損失函數的第一式一拉近同義詞與拉遠反義詞：根據實驗結果，一般常用在調整詞向量的餘弦距離公式與歐幾里得距離相比，反而表現較差。我們拿 [4] 的損失函數與 $Loss_{Eu+MSE}$ 對比，在第二式同樣使用均方差的情況下， $Loss_{Eu+MSE}$ 雖然表現不太穩定，有時候甚至比調整前來的更差，但其在大部分情況下均表現較 [4] 好。若拿 $Loss_{Cos+Eu}$ 及 $Loss_{Eu+Eu}$ 比較，則可以發現， $Loss_{Eu+Eu}$ 均表現較好，且 $Loss_{Cos+Eu}$ 表現均比調整前的差。而第二式的部分一控制調整前後的詞向量的調整程度：本研究所使用的兩種調整方法分別為歐幾里得距離與均方差，而這兩種方法只差在有無做平方根，但兩種損失函數所調整出來的結果卻有相當程度的差別，不管是 [4] 對比 $Loss_{Cos+Eu}$ ，或是 $Loss_{Eu+Eu}$ 對比 $Loss_{Eu+MSE}$ ，第二式在使用均方差的情況下均能獲得較好的結果。

5.2 未來展望

在實驗過後，我們發現由於我們的測試資料為同義詞與反義詞，而詞向量裡的所以字詞並不一定都會出現在這些資料集裡，此也造成訓練過程中只調整了字詞本身與其同反義詞的詞向量，實際上並未調整到其他不包含在同反義詞資料集內的字詞。因此，即使讓同義詞反義詞互相學習到彼此的資訊，但卻影響了原先向量空間中各個字詞之間的資訊。

之後，本研究未來目標以 Word2Vec[3] 的訓練方法—skip-gram 與連續詞袋 (CBOW) 來對文本進行訓練，並加上自注意力機制同時訓練同反義詞，使詞向量除了能學習到文本資訊外，也能學習到詞庫中的其他知識。

參考文獻

- [1] Z. S. Harris, “Distributional structure,” *Word*, vol. 10, no. 2-3, pp. 146–162, 1954.
- [2] J. Pennington, R. Socher, and C. D. Manning, “Glove: Global vectors for word representation,” in *Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1532–1543. [Online]. Available: <http://www.aclweb.org/anthology/D14-1162>.
- [3] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” *arXiv preprint arXiv:1310.4546*, 2013.
- [4] 林冠佑, “基於自注意力機制產生的無方向性序列編碼器使用同義詞與反義詞資訊調整詞向量,” Thesis, 2020. [Online]. Available: <https://hdl.handle.net/11296/qe48u8>.
- [5] W.-t. Yih, G. Zweig, and J. C. Platt, “Polarity inducing latent semantic analysis,” in *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 2012, pp. 1212–1222.
- [6] M. Yu and M. Dredze, “Improving lexical embeddings with semantic knowledge,” in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Baltimore, Maryland: Association for Computational Linguistics, Jun. 2014, pp. 545–550. DOI: 10.3115/v1/P14-2089. [Online]. Available: <https://www.aclweb.org/anthology/P14-2089>.
- [7] C. Xu, Y. Bai, J. Bian, B. Gao, G. Wang, X. Liu, and T.-Y. Liu, “Rc-net: A general framework for incorporating knowledge into word representations,” in *Proceedings of the 23rd ACM international conference on conference on information and knowledge management*, 2014, pp. 1219–1228.
- [8] J. Bian, B. Gao, and T.-Y. Liu, “Knowledge-powered deep learning for word embedding,” in *Joint European conference on machine learning and knowledge discovery in databases*, Springer, 2014, pp. 132–148.
- [9] D. Fried and K. Duh, “Incorporating both distributional and relational semantics in word representations,” *arXiv preprint arXiv:1412.4369*, 2014.

- [10] E. Pavlick, P. Rastogi, J. Ganitkevitch, B. Van Durme, and C. Callison-Burch, “Ppdb 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification,” in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, 2015, pp. 425–430.
- [11] R. Schwartz, R. Reichart, and A. Rappoport, “Symmetric pattern based word embeddings for improved word similarity prediction,” in *Proceedings of the nineteenth conference on computational natural language learning*, 2015, pp. 258–267.
- [12] M. Ono, M. Miwa, and Y. Sasaki, “Word embedding-based antonym detection using thesauri and distributional information,” in *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2015, pp. 984–989.
- [13] D. Osborne, S. Narayan, and S. B. Cohen, “Encoding prior knowledge with eigenword embeddings,” *Transactions of the Association for Computational Linguistics*, vol. 4, pp. 417–430, 2016.
- [14] M. Faruqui, J. Dodge, S. K. Jauhar, C. Dyer, E. Hovy, and N. A. Smith, “Retrofitting word vectors to semantic lexicons,” in *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Denver, Colorado: Association for Computational Linguistics, May 2015, pp. 1606–1615. DOI: 10.3115/v1/N15-1184. [Online]. Available: <https://www.aclweb.org/anthology/N15-1184>.
- [15] N. Mrkšić, D. Ó Séaghdha, B. Thomson, M. Gašić, L. M. Rojas-Barahona, P.-H. Su, D. Vandyke, T.-H. Wen, and S. Young, “Counter-fitting word vectors to linguistic constraints,” in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, San Diego, California: Association for Computational Linguistics, Jun. 2016, pp. 142–148. DOI: 10.18653/v1/N16-1018. [Online]. Available: <https://www.aclweb.org/anthology/N16-1018>.
- [16] J. Wieting, M. Bansal, K. Gimpel, and K. Livescu, “From paraphrase database to compositional paraphrase model and back,” *Transactions of the Association for Computational Linguistics*, vol. 3, pp. 345–358, 2015. DOI: 10.1162/tac1_a_00143. [Online]. Available: <https://www.aclweb.org/anthology/Q15-1025>.
- [17] N. Mrkšić, I. Vulić, D. Ó. Séaghdha, I. Leviant, R. Reichart, M. Gašić, A. Korhonen, and S. Young, “Semantic specialization of distributional word vector spaces using monolingual and cross-lingual constraints,” *Transactions of the association for Computational Linguistics*, vol. 5, pp. 309–324, 2017.

- [18] J.-K. Kim, M.-C. de Marneffe, and E. Fosler-Lussier, “Adjusting word embeddings with semantic intensity orders,” in *Proceedings of the 1st Workshop on Representation Learning for NLP*, Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 62–69. DOI: 10.18653/v1/W16-1607. [Online]. Available: <https://www.aclweb.org/anthology/W16-1607>.
- [19] H. Jo and S. J. Choi, “Extrofitting: Enriching word representation and its vector space with semantic lexicons,” in *Proceedings of The Third Workshop on Representation Learning for NLP*, Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 24–29. DOI: 10.18653/v1/W18-3003. [Online]. Available: <https://www.aclweb.org/anthology/W18-3003>.
- [20] A. V. Oord, N. Kalchbrenner, and K. Kavukcuoglu, “Pixel recurrent neural networks,” in *Proceedings of The 33rd International Conference on Machine Learning*, M. F. Balcan and K. Q. Weinberger, Eds., ser. Proceedings of Machine Learning Research, vol. 48, New York, New York, USA: PMLR, 20–22 Jun 2016, pp. 1747–1756. [Online]. Available: <http://proceedings.mlr.press/v48/oord16.html>.
- [21] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [22] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, “Empirical evaluation of gated recurrent neural networks on sequence modeling,” *arXiv preprint arXiv:1412.3555*, 2014.
- [23] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., Curran Associates, Inc., 2017, pp. 5998–6008. [Online]. Available: <http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf>.
- [24] Z. Yang, Z. Dai, Y. Yang, J. G. Carbonell, R. Salakhutdinov, and Q. V. Le, “Xlnet: Generalized autoregressive pretraining for language understanding,” *CoRR*, vol. abs/1906.08237, 2019. arXiv: 1906.08237. [Online]. Available: <http://arxiv.org/abs/1906.08237>.
- [25] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: pre-training of deep bidirectional transformers for language understanding,” *CoRR*, vol. abs/1810.04805, 2018. arXiv: 1810.04805. [Online]. Available: <http://arxiv.org/abs/1810.04805>.
- [26] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, “Learning word vectors for sentiment analysis,” in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, Portland, Oregon, USA: Association for Computational Linguistics, Jun. 2011, pp. 142–150. [Online]. Available: <http://www.aclweb.org/anthology/P11-1015>.

- [27] F. Morin and Y. Bengio, “Hierarchical probabilistic neural network language model,” in *Aistats*, Citeseer, vol. 5, 2005, pp. 246–252.
- [28] I. Yamada, A. Asai, J. Sakuma, H. Shindo, H. Takeda, Y. Takefuji, and Y. Matsumoto, “Wikipedia2vec: An efficient toolkit for learning and visualizing the embeddings of words and entities from wikipedia,” *arXiv preprint 1812.06280v3*, 2020.
- [29] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, “Enriching word vectors with subword information,” *arXiv preprint arXiv:1607.04606*, 2016.
- [30] E. Pavlick, P. Rastogi, J. Ganitkevitch, B. Van Durme, and C. Callison-Burch, “PPDB 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification,” in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, Beijing, China: Association for Computational Linguistics, Jul. 2015, pp. 425–430. DOI: 10.3115/v1/P15-2070. [Online]. Available: <https://www.aclweb.org/anthology/P15-2070>.
- [31] S. Rajana, C. Callison-Burch, M. Apidianaki, and V. Shwartz, “Learning antonyms with paraphrases and a morphology-aware neural network,” in *Proceedings of the 6th Joint Conference on Lexical and Computational Semantics (*SEM 2017)*, Vancouver, Canada: Association for Computational Linguistics, Aug. 2017, pp. 12–21. DOI: 10.18653/v1/S17-1002. [Online]. Available: <https://www.aclweb.org/anthology/S17-1002>.
- [32] G. A. Miller, “Wordnet: A lexical database for english,” *Commun. ACM*, vol. 38, no. 11, pp. 39–41, Nov. 1995, ISSN: 0001-0782. DOI: 10.1145/219717.219748. [Online]. Available: <https://doi.org/10.1145/219717.219748>.
- [33] C. F. Baker, C. J. Fillmore, and J. B. Lowe, “The Berkeley FrameNet project,” in *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, Montreal, Quebec, Canada: Association for Computational Linguistics, Aug. 1998, pp. 86–90. DOI: 10.3115/980845.980860. [Online]. Available: <https://www.aclweb.org/anthology/P98-1013>.
- [34] F. Hill, R. Reichart, and A. Korhonen, “SimLex-999: Evaluating semantic models with (genuine) similarity estimation,” *Computational Linguistics*, vol. 41, no. 4, pp. 665–695, Dec. 2015. DOI: 10.1162/COLI_a_00237. [Online]. Available: <https://www.aclweb.org/anthology/J15-4004>.
- [35] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

附錄 A 實驗程式碼

本研究的實驗程式碼公開於 Github 軟體原始碼代管服務平台

連結網址:<https://github.com/ncu-dart/Adjusting-Word-Embeddings-and-Downstream-Tasks>

詳細操作方法請參閱 README.md 文件