中　山　醫　學　大　學

醫 學 資 訊 學 系

畢業專題文件

EDM-RoBERTa : Enhancing the Dependency Mechanism of RoBERTa

專題編號:

專題學生: 李昱廷　郭為軒　曹仲辰
吳岳霖　林裕峰

指導教授: 張炎清 博士

中　華　民　國　一　百　一　十　年　元　月

摘要

本模型 EDM-RoBERTa (Enhancing the Dependency Mechanism of RoBERTa) 是以具單頭注意力的遞迴神經網路 SHA-RNN (Single-headed Attention Recurr- ent Neural Networks) 改良 Transformer 編碼器中的多頭注意力機制 (Multi-headed Attention) 。研究來源以基於 Transformer 的模型(包含 BERT、RoBERTa、XLNet、DistilBERT)、SHA-RNN 及 Transformer 中的自注意力機制為主軸進行研究。研究中所使用的方法將透過 SHA-RNN 的 Boom Layer 改造後的遞迴神經網路實現注意力機制以進行高維度向量轉換，改良 Transformer 編碼器中原有的多頭注意力機制將解決 Transformer 在輸入序列中弱於捕獲局部文本依賴關係中產生的短期依賴問題。

我們提出新的模型架構 EDM-RoBERTa 透過 Boom Layer 與原始 Transformer 編碼器 RoBERTa，重組架構。與原本單獨的 Transformer 和 SHA-RNN 模型相比能同時滿足長文本序列輸入所需的短期依賴及具備 Transformer 原有的長期依賴特點。在運算過程中亦能透過減少計算量以提升精度及文本分類表現。本研究所獲得的成果將輔助應用於情感分析、社交網路分析、聊天機器人及疾病傳播預測。

關鍵字: 情感分析、RNN、Transformer、單頭注意力、SHA-RNN、Boom Layer

一、 研究動機與研究問題

◆ 研究動機

自然語言處理是結合人工智慧和語言學領域的重要方向，注重於自然語言與電腦之間的通訊互動。主要包含自然語言理解及自然語言生成，其中自然語言理解包含從自然語言的表達及語句中識別出該句真正的目的及含義。然而，機器理解過程則因段落句子組合不同而識別出不同的含義，包含一詞多義造成的理解錯誤。現行模型訓練主要以自然語言理解 (NLU, Natural Language Understanding) 發展預訓練模型 (Pre-trained Model)，用以輔助人機之間的溝通理解及後續的自然語言生成 (NLG, Natural Language Generation)，因此語義特徵的提取能力尤為重要。在自然語言生成部分，語言模型有能夠完成問答、閱讀理解、段落總結的能力。

以遞迴神經網路 (RNN, Recurrent Neural Networks) 做特徵提取，在提取過程中，詞依時序讀入被分配不同的權重。隨著詞與詞之間的距離拉遠及網路越深，先前被輸入詞的權重會被稀釋，造成包含前面被輸入的資訊量會越來越少。透過 Transformer 作為主要之特徵提取模型，避免先後輸入造成的權重稀釋問題，與透過多頭注意力機制針對當前預測詞，將同時用到前面和後面的詞進行計算。Transformer 透過平行計算提升訓練及計算效率，相較之下 RNN 將因時序複雜度太大導致計算效率低下。

Transformer 與 RNN 在單詞處理主要差別在於時序輸入，Transformer 的無序輸入會造成段落句子及單詞組合不同而識別出不同的含義，此時加入位置編碼 (Position Encoding) ，將位置編碼與詞嵌入向量內積作為輸入的嵌入向量，使詞向量包含位置訊息。在輸入過程中包含將輸入的嵌入向量透過多層的多頭注意力機制、前饋層 (Feed-Forward Neural Network)、與層標準化 (Layer Normalization) 。輸出過程包含以殘差連接連接多層的多頭注意力與前饋層。

Transformer 的特徵提取技術在研究中已相當成熟，其中的注意力機制的研究亦不斷透過改善以提升運算效能。因此，以 SHA-RNN (Single-Headed Attention RNN)實作注意力，達到單頭注意力的技術受到矚目。

Transformer 的架構中用到多頭注意力 (Multi-Head Attention)，避免順序輸入的缺失並提高預測效率，但無法確認每層的多頭中幾個為有效頭數。此外，Transformer 主要以在計算過程中的記憶體瓶頸，與短期文本無可避免的短期依賴問題。

相較而言，SHA-RNN 的注意力在每層隱藏層中只保留單個頭，過程中減少多餘的運算量。因此，為了提升運算效能，本研究藉由 SHA-RNN 中的最佳化技術 Boom Layer 與 Transformer 中的編碼器進行修改重組 Transformer 結構執行自然語言文本分類任務。

◆ 研究問題

Transformer 的運作過程因為會給予單一句子中出現的重複詞相同的權重，會造成無法給予相對鄰近詞有較大的權重，進而在自然語言理解也會出現問題。研究中藉由 Transformer 及 SHA-RNN 重組編碼器和解碼器結構，能夠解決 Transformer 弱於捕獲短期文本的依賴問題。透過結合 SHA-RNN 技術能有助於解決 Transformer 弱於捕獲短期文本依賴問題，且能夠以單頭注意力提升運算效率。

本研究的主題如下，重組後的編碼器與進行更準確的自然語言文本分類任務。首先，對輸入語句以基於 Transformer 的模型(BERT、RoBERTa、XLNet、DistilBERT)進行預訓練，包含詞塊化 (Tokenization)、文本清理 (Text Cleaning) 以及模型訓練。訓練過程包含判斷兩句話是否有相同的含義，並捕捉句子之間的關係。透過 SHA-RNN 降低運算量，注意力機制的過程以 LSTM 實作單頭注意力解決 Transformer 短期文本依賴問題，需要對向量進行矩陣乘法，該方法與傳統的下映射層相比能減少整個矩陣運算量。

因此，在研究中將對 Transformer 的編碼器、SHA-RNN 注意力機制與多頭注意力機制之不同、Transformer 與 SHA-RNN 的融合效果進行分析和探討。

二、文獻回顧與探討

一、 Transformer

Transformer 是一種基於自注意力機制的Seq2Seq(Sequence to Sequence) 模型，常用於提高神經網路機器翻譯的性能及模型訓練的速度，在特定任務中的表現優於傳統的機器翻譯模型。

✧ 自注意力機制 (Self-Attention)

Transformer編碼器的輸入首先經過自注意力層，這幫助編碼器對特定單詞進行編碼的同時也查看輸入語句的其他單詞，接著自注意力層的輸出會送到前饋神經網路。

解碼器包含自注意力機制、前饋層、一層編碼解碼的注意力層，協助解碼器將注意力集中於輸入語句的相關部分，如圖1。

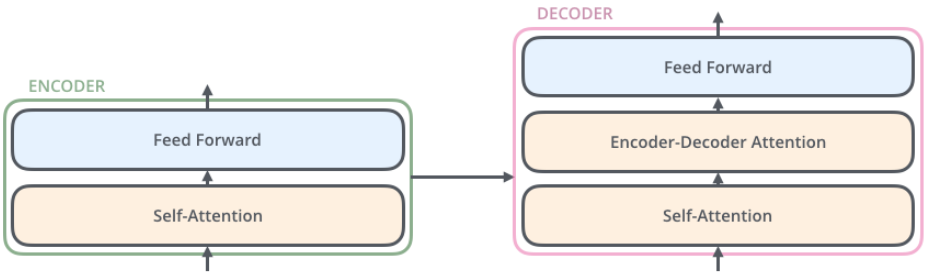


圖 1 , 編碼器與解碼器架構圖, 來源: jalammer.github.io

輸入語句的每個單詞將建立三個向量 (query查詢向量、key鍵向量、value值向量) ，透過將單詞的嵌入向量對訓練過程建立的三個矩陣 (query, key, value) 進行點乘 (Dot-Production) 產生向量。

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

注意力得分分別透過查詢向量之點乘與各個單詞鍵向量內積得出，分別為q1·kn、q2·kn……分別除以向量維度的平方根以獲得更穩定之梯度。

透過softmax對分數進行標準化，顯示該單詞對目標單詞的相關性。將每個值向量乘以softmax分數，並加權總和就能得到自注意力輸出，如圖2。

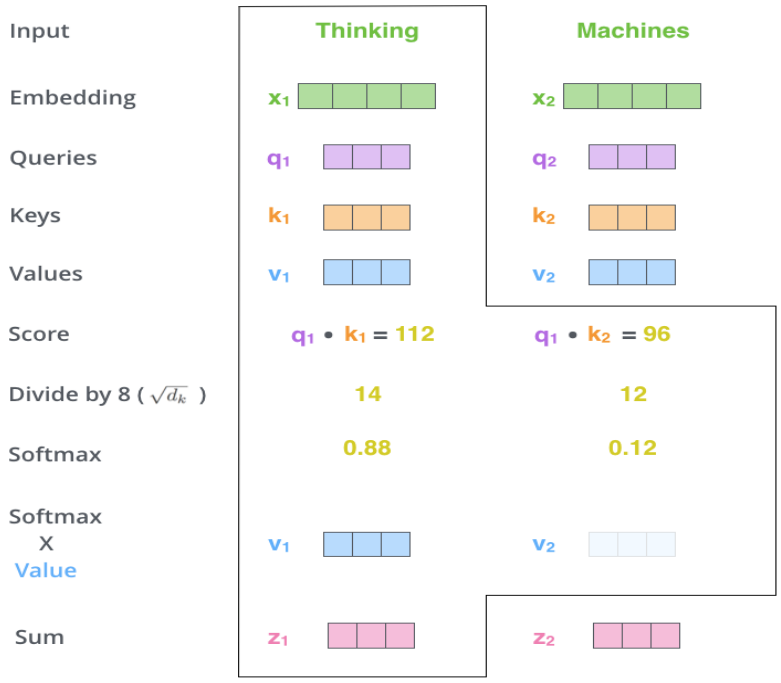


圖 2, 自注意力結構圖, 來源: jalammer.github.io

✧ 多頭注意力機制 (Multi-Head Attention)

多頭注意力為單頭注意力的原理延伸，多頭注意力著重於單一查詢矩陣和多個鍵向量進行點乘並一起考慮整個輸入語句的單詞，如圖3。

根據不同下游任務不同的頭關注的點不同，其中包含以單頭取局部的資訊或以多頭取全局的資訊。因此，無論是單頭注意力機制或多頭注意力機制，在特定問題的解決方案都同時被用到，如圖4。

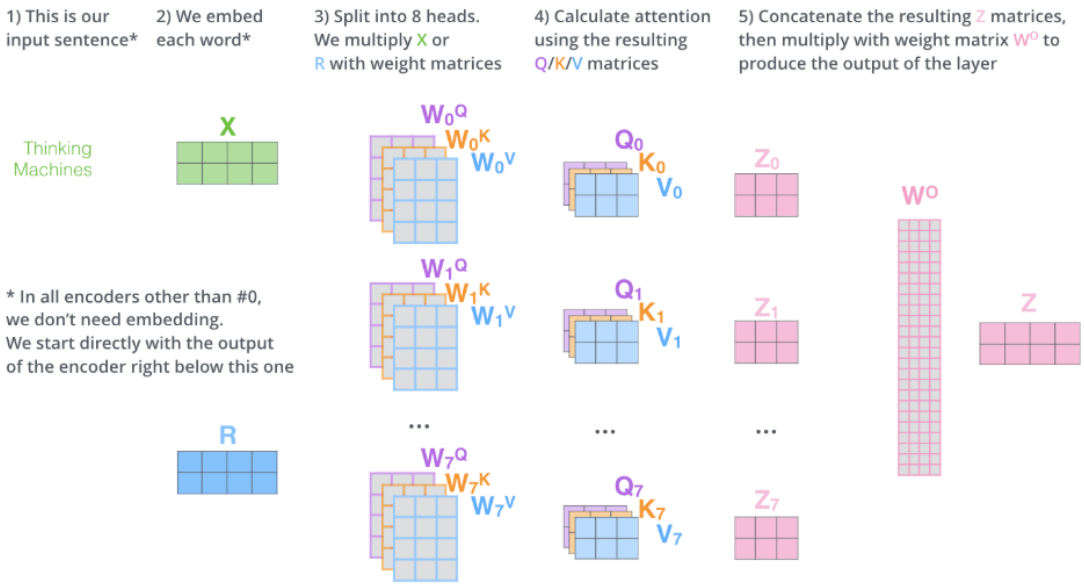


圖 3, 多頭注意力結構圖, 來源: jalammer.github.io

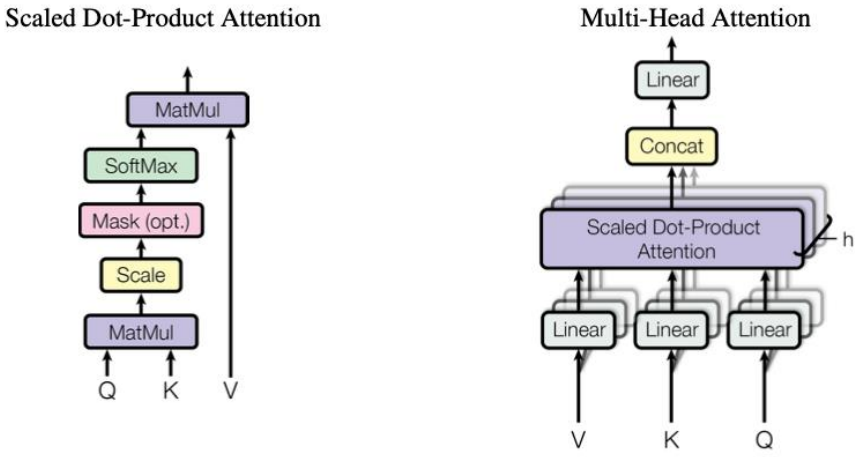


圖4, 注意力機制比較圖, 來源: Attention Is All You Need 論文

編碼過程中，與傳統 RNN 相比確實能夠降低順序輸入問題，但也因為無序問題而衍生出位置編碼 (Position Encoding) 之必要性。在未加入位置編碼情況下，自注意力機制會將句子中出現重複的詞賦予一樣的權重。如此將造成相近之詞應較重要，但權重卻與另一個較遠之詞一樣重要，衍伸出 Transformer 弱於捕獲文本中之短期依賴的問題，這種對注意力的依賴可能會導致 Transformer 於語法敏感任務上之性能不

如 RNN 模型。

因此，RNN 模型過度仰賴短期依賴，Transformer 缺少必要之短期依賴。

二、 Bidirectional Encoder Representations from Transformers (BERT)

✧ 預訓練 (Pre-Training) 雙向 Transformer

傳統的語言模型因由數學定義為單向而且LSTM只能完成淺層訓練，導致對於不同位置方向的單詞而言，在編碼的過程看不到另一側的單詞。雖然句子中有些單詞會依賴鄰近左右側的單詞，但僅僅從單方向做編碼無法滿足需求。

透過 Transformer進行自然語言處理任務，與RNN不同之處，在於能將網路做得更深，不同位置之詞都能不受位置距離和方向因素而進行編碼。然而，在語言生成之組合部分， 即使BERT做詞嵌入時有加入位置編碼 (Position Encoding) ，但其原理是被用來與輸入嵌入求平均，因此語言組合也涉及詞序推理，並非僅需注意力機制 (Attention *isn't* all you need.)。

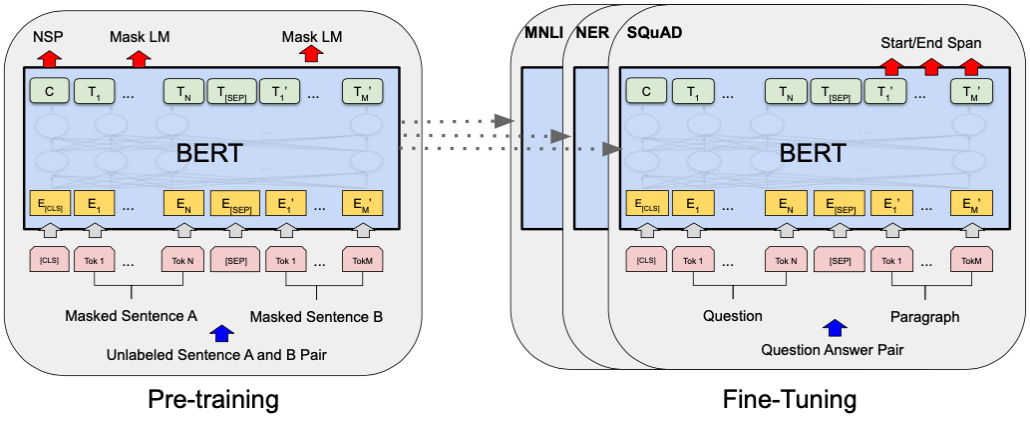


圖 4,BERT預訓練及微調結構圖, 來源: BERT 論文

✧ 預訓練任務 #1 Masked Language Model (Masked LM)

訓練語言模型，將大量未標註之數據進行無監督學習，使其學習語法結構、解讀語義，並透過語言模型減少不同自然語言處理任務之預訓練和建構成本。

在預訓練之前對訓練集隨機遮蔽 15% 的單詞，而非如以往將每個詞都預測一次。最後，損失函數只會計算被遮蔽之詞塊 (token) ，被遮蔽的15%中有10%被替換成其他單詞，另外 10% 不替換，剩餘 80% 被替換為 [MASK] 。

預訓練過程中，模型將猜測 15% 所有詞塊，對每個詞均計算損失與注意力機制，易造成當 [MASK] 出現過多，影響模型收斂速度，甚至比 RNN 左到右的模型更慢。

✧ 預訓練任務 #2 Next Sentence Prediction (NSP)

判斷第二個語句在原始文本中是否為第一句子之後續語句。

✧ 現今 NLP 兩階段遷移學習

以語言模型預訓練方法訓練出對自然語言有相當程度理解之語言模型，並將其用以做特徵擷取並針對下游任務進行微調。透過 BERT 將同時完成無監督學習和監督式微調的部分。

三、 Sequence to Sequence (Seq2Seq)

✧ 編碼器與解碼器

Seq2Seq 模型主要由編碼器與解碼器兩個RNN組成，編碼器負責將輸入序列編碼轉換成中間向量 (Context Vector)，解碼器再根據中間向量轉換成文

字輸出。在預測的過程中，目前字詞的預測不僅取決於前面已翻譯的字詞，亦考慮原始輸入。

運作過程中，編碼器最後時間神經元的隱藏層輸出到解碼器的第一個神經元，透過激勵函數與 Softmax 層，篩選出機率最大者做為下一個神經元的輸入，如圖5。

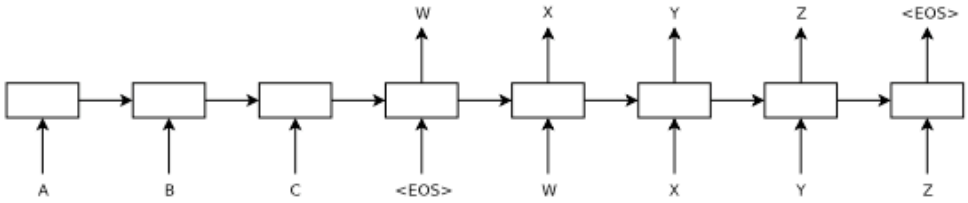
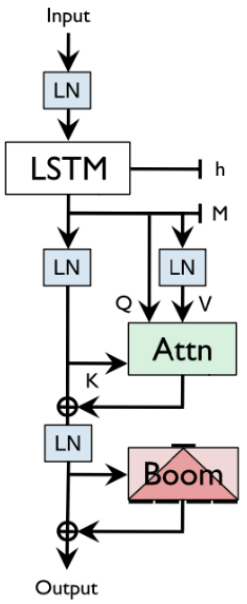


圖 5,Seq2Seq 結構圖, 來源: Seq2Seq 論文

相關問題出現於中間向量，在編碼器以最後一個神經元進行轉換時，依序由左到右讀取資訊，但中間向量仍為固定維度的向量。導致轉換後的向量無法涵蓋所有輸入序列的訊息，先被輸入的重要訊息將在轉換後權重會變低或甚至消失。

四、 Single Headed Attention RNN (SHA-RNN)



✧ 單頭注意力 (Single Headed Attention)

Transformer 模型建立於無序基礎，並只透過注意力機制完成訓練，但每層網路都有幾十個注意力頭 (Head)，運算的過程中將因無法得知哪些頭為有效而耗費多餘的運算資源。

相較而言，SHA-RNN的注意力機制只保留一個頭完成向量的注意力點乘。

✧ 結構變更

主要以 Transformer 之自注意力機制為基礎修改，四層之結構中，每層先做LSTM，進行層標準化 (Layer Normalization) 後連接注意力機制，因此實際上為 8 層網路。

結構變更，與 Transformer 不同之處在於只對Q做全連接層，以sigmoid產生Q, K。然而，過程中發現 LSTM 輸出須經過全連接層轉換為Q,K,V，因此改進的核心部分在於作者提出改造後的前饋層(Boom Layer)。

✧ Boom Layer

為了減少運算量，基於Transformer前饋層改造後之 Boom Layer，將原本前饋層透過激勵函數GeLU轉換成 N 倍向量，分成 N 等分後透過加總將維度轉換回原維度。透過實作將節省顯存用量，跑更多層網路，使模型降低整個矩陣的運算量，提高模型於單一GPU上訓練的效率。

$$v \in \mathbb{R}^{H} \longrightarrow u \in \mathbb{R}^{N \times H} \longrightarrow w \in \mathbb{R}^{H}$$

<< 維度轉換示意圖

五、RoBERTa: A Robustly Optimized BERT Pretraining Approach

✧ RoBERTa 主要修改

與 BERT 相比，將訓練過程做一部分修改。增大 batch size 與動態遮罩 (Dynamic Masking)，並用更多語料進行訓練。其中修改 BERT 之主因在於其自身訓練不足，因此 RoBERTa 透過多種不同最佳化方法改善 BERT 性能。

✧ 靜態遮罩與動態遮罩

原始 BERT 於資料預處理 (Data Preprocessing) 即使用遮罩，而 RoBERTa 為了避免不同 epochs 都使用到相同之遮罩，提出 10 種不同遮罩。由於共訓練 40 epochs，每筆資料會用到 4 次相同之遮罩。

✧ 動態遮罩

動態遮罩於模型訓練之前，動態生成遮罩模式 (Masking Pattern)，實驗顯示使用動態遮罩的訓練效能比靜態遮罩更好。

✧ 更大 batches

BERT 原始論文中使用的 batch size 為 256，RoBERTa 分別使用 2K 及 8K，減少運算之 steps。此外，更大之 batch-size 更容易進行平行化運算。實驗過程發現，提高 batch size 至 8K，能減低Masked Language Model 之 困惑度 (Perplexity)。

三、資料集來源

資料集蒐集目的與預期結果旨在透過多維度的公開資料微調及訓練模型，提升模型在跨資料集和其他情感分析的泛化能力。此外，透過二分類的情感分析資料集提升模型對於強烈情緒語句之文本分類能力。

✧ SST-5 Fine-grained classification

本資料集為 SST-5 Fine-grained classification 電影評論情感分類集，其中情感標籤包含五種等級:

0 - negative
1 - somewhat negative
2 - neutral
3 - somewhat positive
4 - positive

研究過程透過該資料集訓練不僅提供多維度情感分析，細粒度的情感標籤亦在訓練過中降低模型可能衍伸之情感等級模糊問題。

✧ SST2: Stanford: IMDb Dataset of 50K Movie Reviews

本資料集包含五萬餘則 IMDb 電影評論情感分類集，情感標籤以正面及負面兩極情感維度。訓練集及測試集的文本序列數量分別各為25,000餘則。

✧ Twitter

四、研究方法及步驟

◆ 以 **BERT** 對原始文本進行預訓練與針對不同下游任務進行微調

BERT運作過程主要基於未標註或只有少量標註之文本數據進行微調以解決新的下游任務。運作過程包括以下三個主要步驟:

1. 準備原始文本數據:

文本數據包含未標註或少量標註之文本，透過數據清理將文本中空白標題的範例去除。同時，將超出BERT模型中預設序列長度的範本去除，並以0將將小於序列長度的向量補 0，以符合預訓練文本讀入。

2. 將原始文本轉換成BERT相容之輸入格式:

文本進行預處理過程中對句子開頭向量位置加入分類符 [CLS]，並以[SEP]以 0/1 區分第一句與第二句。再以中文 BERT 對文本進行斷詞，如圖5。

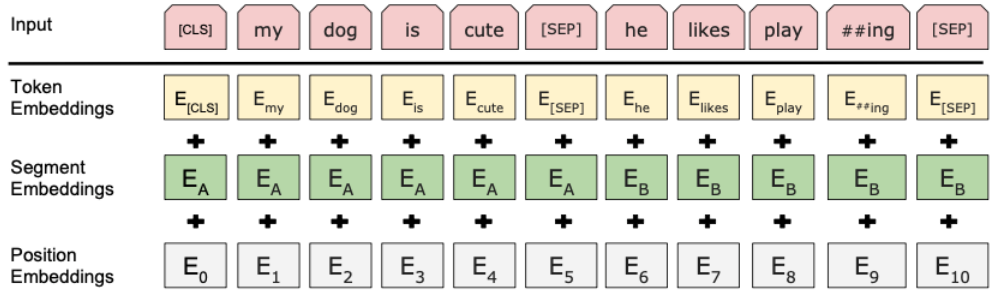


圖 5,BERT 成對句子編碼示意圖, 來源: BERT 論文

3. 於BERT頂層加入新的Layer進行微調使其適用於特定下游任務:

對BERT模型進行微調的部分包含利用下游任務的目標函式從頭訓練分類器並微調BERT參數，以訓練完之BERT加上線性分類器最大化當前下游任務的目標，如圖6。

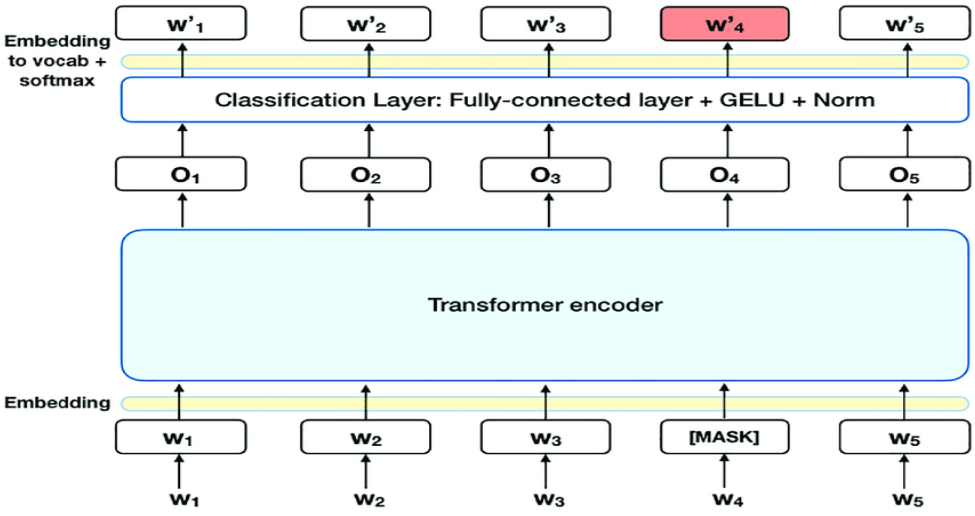


圖 6,BERT分類層示意圖, 來源: Faiza Khattak

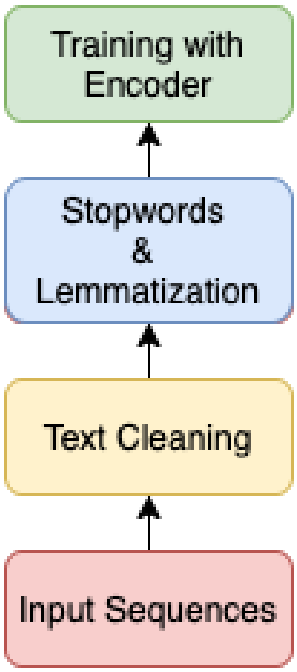
透過遷移學習，新增的分類器大多數的參數都來自已經預訓練的BERT，因此實際上需要從頭訓練的參數量較少。在微調的過程，將依不同下游任務加入不同之線性分類器。

◆ **實驗步驟**

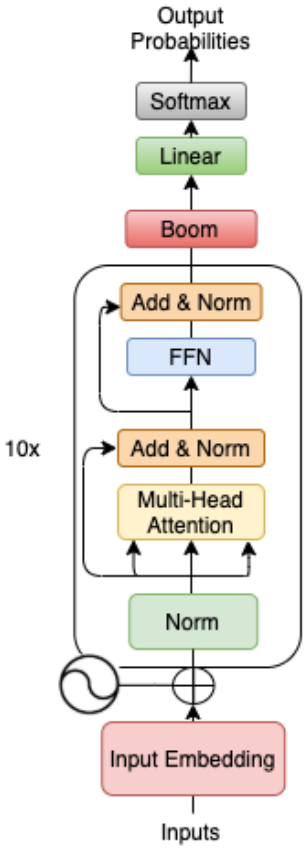
基於Transformer之模型中，挑選於情感分類資料集表現最佳的模型做為主要分類器及修改模型

實驗模型主要以基於Transformer的四種模型進行比較，其中包含 BERT, RoBERTa, XLNet, DistilBERT 對情感分類數據集做分類。

實驗過程中挑選效能最佳之 RoBERTa 做為主要的模型修改，並以Boom Layer做高維度轉換提取更多的文本特徵以及降低整個矩陣運算量。



對文本數據進行預處理



EDM-RoBERTa 模型架構圖

Fine-tuning Transformer-based Models with IMDb Dataset

	Epoch	Accuracy	train loss	valid loss	error rates
$BERT_{LARGE}$	6	92.6	0.35	0.55	0.29
$RoBERTa_{LARGE}$	6	93.17	0.22	0.53	0.26
XLNet	6	89.53	0.28	0.69	0.37
DistilBERT	6	86.48	0.32	0.74	0.35
EDM-RoBERTa	6	94.76	0.27	0.49	0.2

<<以實驗樣本對 IMDb Dataset 做二分類情感分類

Fine-tuning Transformer-based Models with Rotten Tomatoes Dataset

	Epoch	Accuracy	train loss	valid loss	error rates
$BERT_{LARGE}$	5	66.21	0.64	0.68	0.3
$RoBERTa_{LARGE}$	5	68.91	0.67	0.7	0.29
XLNet	5	62.83	0.73	0.79	0.38
DistilBERT	5	54.65	0.8	0.77	0.44
EDM-RoBERTa	5	76.18	0.64	0.62	0.26

<<以實驗樣本對 Rotten Tomatoes Dataset 做多維度情感分類 (此資料集為 5 維度)

EDM-RoBERTa (Enhance the Dependency Mechanism of RoBERTa)

bsz	steps	lr	ppl	SST-2	SST-5
256	1M	1.00E-05	3.83	92.6	74.57
2K	**125K**	**2.00E-04**	**3.61**	**94.76**	**76.18**
8K	31K	1.00E-03	3.72	92.1	74.31

<< EDM-RoBERTa 之訓練細節與效能比較

實驗過程分別以 SST-2 與 SST-5 資料集驗證模型效能，實驗環境主要包含 Google Colaboratory (GPU: Nvidia V100, 16GB) 與 Intel Core i9-9980HK with AMD Radeon Pro 5600M 8GB HBM2 進行訓練。期間以 Wiki-Text 與 CC-NEWs 語料對 EDM-RoBERTa 進行預訓練，後續以本模型對 SST-3 及 SST-5 進行監督式微調，以完成模型訓練。

實驗數據顯示，我們提出的EDM-RoBERTa於二維度與多維度的文本分類表現皆優於其他包含BERT的模型，誤差亦更低。在驗證模型步驟中，batch size以2K及學習率 2e-04 獲得良好的分類表現。因此，實驗證明透過比BERT更大的batch訓練以及較大的學習率，模型在SST-2與SST-5均能獲得更好之效能表現。

五、討論

經由研究過程，EDM-RoBERTa 將獲得由原始文本經 RoBERTa 進行預訓練及監督式微調的結果，透過輸出並以 Boom Layer 以單頭注意力完成編碼器的前饋層及高維度向量轉換，最後得出比原始 Transformer 詞義分析輸出更精準的預測結果，改善 Transformer 注意力機制導致編解碼過程無序的缺失以及其衍生出的弱於捕獲文本中的短期依賴問題。最後將模型應用於社交網路分析、情感機器人、商品評價分析與後續其他自然語言情感分析任務。

六、參考文獻

1. Jacob Devlin, Ming-Wei Chang, Kenton Lee and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv: 1810.04805v2, 2019.
2. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jacob Uszkoreit, Llion Jones, Aidan N.Gomez, Lukasz Kaiser and Illia Polosukhin. Attention Is All You Need. arXiv: 1706.03762v5, 2017.
3. Yoav Goldberg. Assessing BERT’s Syntactic Abilities. arXiv: 1901.05287v1, 2019.
4. Richard Socher, Alex Perelygin, Jean Y. Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng and Christopher Potts. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank, 2013.

5. Stephen Merity. Single Headed Attention RNN: Stop Thinking With Your Head. arXiv: 1911.11423v2, 2019.
6. Ilya Sutskever, Orial Vinyals and Quo V.Le. Sequence to Sequence Learning with Neural Networks. arXiv: 1409.3215v3, 2014.
7. Nikita Kitaev, Lukasz Kaiser and Anselm Levskaya. Reformer: The Efficient Transformer. arXiv: 2001.04451v2, 2020.
8. Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V.Le and Mohammad Norozui. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. arXiv: 1609.08144v2, 2016.
9. Stephen Merity, Caiming Xiong, James Bradbury and Richard Socher. Pointer Sentinel Mixture Models. arXiv: 1609.07843v1, 2016.
10. Lajanugen Logeswaran and Honglak Lee. An Efficient Framework for Learning Sentence Representation. arXiv: 1803.02893v1, 2018.