The 2019 Conference on Computational Linguistics and Speech Processing
ROCLING 2019, pp. 3-12

新穎的序列生成架構於中文重寫式摘要之研究

Novel Sequence Generation Framework for Chinese Abstractive Summarization

簡靖岳　Chin-Yueh Chien
國立臺灣科技大學資訊工程系
Department of Computer Science and Information Engineering
National Taiwan University of Science and Technology
M10615110@mail.ntust.edu.tw

陳冠宇　Kuan-Yu Chen
國立臺灣科技大學資訊工程系
Department of Computer Science and Information Engineering
National Taiwan University of Science and Technology
kychen@mail.ntust.edu.tw

摘要

近年網路訊息的爆發式成長，人們每天都能接觸到海量的訊息，但文章中常常包含非必要的資訊、雜訊，降低人們閱讀的效率，若能使用自動文章摘要(Automatic Document Summarization)的技術，將文章中重點萃取出來，便可大幅節省人們閱讀的時間。目前的自動摘要方法，主要分為抽取式(Extractive)摘要與重寫式(Abstractive)摘要，且大部分研究皆驗證在英文的資料集上。本論文提出兩種新穎的序列生成架構於重寫式摘要，包含「以 BERT 為編碼器之指針生成摘要法」與「融合 BERT 與 Transformer 之指針生成摘要法」。此外，目前重寫式摘要研究多半是以英文語料為研究目標，因此在本研究中，我們探討這些模型於中文重寫式摘要的任務成效，以作為後續研究的重要比較基礎。

關鍵詞：自動文章摘要、BERT、Transformer、指針生成網路。

一、緒論

自動文章摘要之研究分為兩大類，抽取式(Extractive)摘要與重寫式(Abstractive)摘要。前者依據特定的摘要比例，從原始文章中選取具代表性的語句來組成摘要。後者則是讓機器閱讀整篇文章，理解文章內容後，重新撰寫摘要代表這篇文章，其使用的詞彙不一定

全來自原始文章，這種摘要方式可說是更貼近人類平常撰寫摘要的形式。

近年來自動摘要的研究中，序列對序列模型應用於重寫式摘要的研究[1-5]，在眾多資料集中驗證其豐碩的成果。特別是近年提出的指針生成網路(Pointer Generator Network, PGN)，其機制可以有效解決文章中存在非字典詞彙(Out-of-vocabulary)的問題，因此被應用在重寫式摘要上。近年由谷歌提出的 Transformer[6]架構，使用注意力(Attention)機制，可以解決遞迴神經網路的長時間序列信息丟失問題並平行處理，加速網運算。谷歌基於 Transformer 架構進一步地提出 BERT，在自然語言處理(Natural Language Processing, NLP)的各項任務中，使用 BERT 效果均獲得顯著的提升。

本論文提出兩個新穎的重寫式摘要模型。第一個模型以指針生成網路為基礎，加上 BERT 作為編碼器，期望 BERT 能生成強健且準確的文章表示特徵，提升重寫式摘要的任務成效，我們稱為「以 BERT 為編碼器之指針生成摘要法」。第二個模型為第一個模型的延伸，除了使用 BERT 作為編碼器外，我們使用 Transformer 架構代替傳統遞迴神經網路，讓注意力機制來獲得時間序列上的關係，用以解決長時間序列訊息丟失問題，產生更加強健且依賴上下文資訊的特徵，期望讓重寫式摘要效能再進一步提升，我們稱為「融合 BERT 與 Transformer 之指針生成摘要法」。此外，我們將探討這些模型在中文重寫式摘要的任務成效，以作為後續研究的重要比較基礎。

二、相關方法

（一）序列對序列模型

序列對序列模型(Sequence-to-sequence)[7]主要包含兩大部分，編碼器(Encoder)與解碼器(Decoder)。編碼器與解碼器大多由遞迴神經網路(Recurrent Neural Networks)構成，例如長短期記憶網路(Long Short-term Memory, LSTM)[8]。

給定一段詞彙序列$\{w_1, w_2 \dots, w_n, \dots w_N\}$依序輸入編碼器中，每個詞彙$w_N$之詞向量$x_n$會與前一個時間點遞迴神經網路的輸出$h_{n-1}$，一起輸入遞迴神經網路，產生此時間點遞迴神經網路的輸出h_n。在解碼器部分，由於輸入文章的每個詞彙對於解碼器產生的每個輸出重要程度並不一樣，有研究提出加入注意力機制[9]，使得解碼器在每個時間點，會對編碼器的所有時間點產生一個注意力權重，表示編碼器中每一個時間點對於此時解碼

器的重要性。在解碼器中，遞迴神經網路會在每個時間點產生一個輸出向量s_t，此一向量將與編碼器的每個時間點輸出h_n計算得到一個相關性權重e_n^t，並透過正規化(Normalize)，獲得t時間點，解碼器對於編碼器的注意力權重$a^t = softmax(e_n^t)$，其中v^T, W_h, W_s, b_{attn}即為注意力機制中的模型參數。接著，將每一個注意力權重a_n^t與所對應的h_n相乘後加總，就可獲得當前時間點之注意力向量s_t^*:

$$e_n^t = v^T tanh(W_h h_n + W_s s_t + b_{attn}) \quad \text{式(1)}$$

$$s_t^* = \sum_{n=1}^{N} a_n^t h_n \quad \text{式(2)}$$

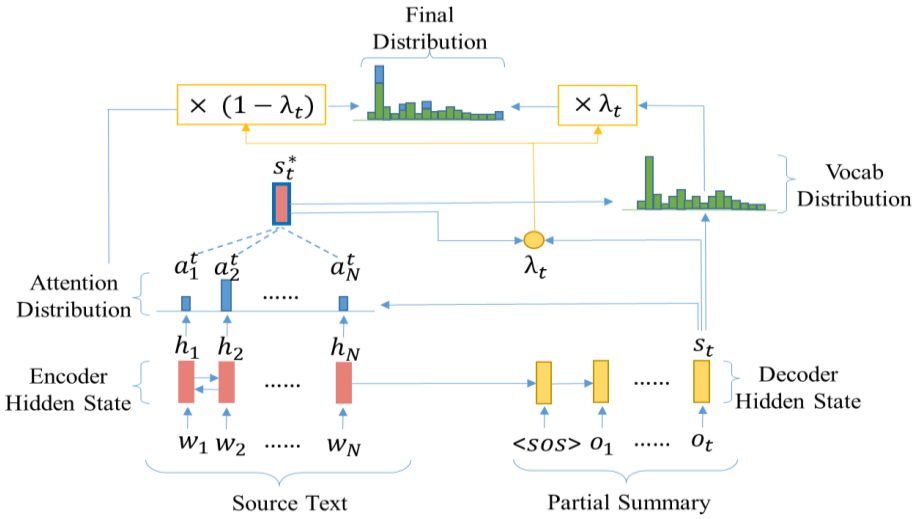
再經過全連接層以及$softmax$激活函數，會輸出一個辭典大小維度的機率分布，每一個維度對應一個字典中的詞。我們將分數最高的詞選取出來，作為此時間點的輸出，同時也是下個時間點的輸入。重複步驟直到解碼器輸出特殊符號 $< EOS >$為止。

（二）指針生成網路(Pointer Generator Network, PGN)

在傳統序列對序列之重寫式摘要的任務中，某些詞彙雖然出現在輸入文字序列中，但若它並非存在字典中(Out-of-Vocabulary, OOV)，這類詞彙是無法被解碼器解碼出來的。為了解決此一問題，近年有研究提出了指針生成網路(Pointer Generator Network, PGN)，其架構如圖一所示。一篇文章之詞彙$\{w_1, w_2 \dots, w_n, \dots w_N\}$，逐一輸入編碼器遞迴神經網路後，產生每個時間點的輸出向量h_n。與序列對序列模型中的解碼器一樣，遞迴神經網路會在每個時間點產生輸出s_t，並與編碼器每個時間點h_n產生e_n^t，藉由正規化得到注意力權重a^t，然後我們將注意力權重$a^t = [a_1^t, \dots, a_n^t, \dots, a_N^t]$乘上編碼器每個時間點之$h_n$，相加後得到當前時間點解碼器之注意力向量$s_t^*$，再透過全連接層以及$softmax$激活函數，可以得到字典(Vocabulary)中所有詞彙的機率分布P_{vocab}：

$$P_{vocab} = softmax(\grave{V}(V[s_t, s_t^*] + b) + \grave{b}) \quad \text{式(3)}$$

其中，$\grave{V}, V, b, \grave{b}$為可學習之模型參數。除了$P_{vocab}$外，指針生成網路會產生一個僅由輸入的文字序列計算而得的語言模型P_{PGN}，這個語言模型的辭典僅由輸入中所有不同的字詞所組成，因此可能包含沒有出現在P_{vocab}中的字詞。P_{PGN}可以快速地由注意力權重a^t計算而得，由於注意力權重a^t已經過正規化，因此P_{PGN}必定滿足機率公設$\sum_w P_{PGN}(w) = 1$。最後，我們利用s_t、s_t^*和解碼器時間點t之輸入x_t計算出P_{vocab}與P_{PGN}的結合係數λ_t，並



圖一、指針生成網路(Pointer Generator Network, PGN)架構圖

透過線性組合，產生解碼器在時間點t的參考機率分布$P(w)$。藉由此方式，不同於序列對序列模型，指針生成網路便可以輸出非字典裡的詞。

$$P_{PGN}(w) = \sum_{n:\ w_n=w} a_n^t \quad \text{式(4)}$$

$$\lambda_t = \sigma(W_{s^*} s_t^* + W_s s_t + w_x x_t + b_{ptr}) \quad \text{式(5)}$$

$$P(w) = (1-\lambda_t)P_{vocab}(w) + \lambda_t P_{PGN}(w) \quad \text{式(6)}$$

（三）Transformer

Transformer 為谷歌近年提出的序列對序列模型[6]，其使用注意力機制搭配捲積神經網路來處理時間序列輸入。在編碼器端，Transformer 透過矩陣運算的方式，將每個輸入詞彙w_n分別乘上Q, K, V三個矩陣，得到q_n, k_n, v_n三個向量，接著詞彙w_n的q_n向量分別會與所有詞彙的k_i向量計算出注意力權重，而後注意力權重再與所對應的v_i相乘後相加，產生詞彙w_n新的向量表示法。於解碼器端，解碼器的詞彙除了與解碼器其他詞彙計算注意力外，也與編碼器每個輸入詞彙計算注意力，藉此來與編碼器產生關係，最後產生當前時間點之輸出。相較於傳統序列對序列模型，Transformer 主要由注意力機制組成，且因為 Transformer 可以進行平行化訓練，相較於遞迴神經網路，可減少相當多的時間成本。

（四）BERT

基於 Transformer，谷歌又進一步地提出了 BERT(Bidirectional Encoder Representations

from Transformers)模型[10]，它使用多層 Transformer 作為主要模型架構，並在兩個任務上進行訓練。第一個任務是遮蔽式語言模型(Masked Language Model)，其作法是隨機將輸入遮蔽，模型訓練的目標為預測被遮蔽的輸入；第二個任務則是下一句預測(Next Sentence Prediction)，做法為同時輸入兩個句子，模型需預測這兩個句子是否為上下文的關係。BERT 的訓練不需要標記資料，因此它可以在非常大量的資料中進行這兩個任務的訓練，而形成一個強健的預訓練模型，而後，其他自然語言任務只需要在這個預訓練模型上進行參數微調，便能取得很好的效果[11]。

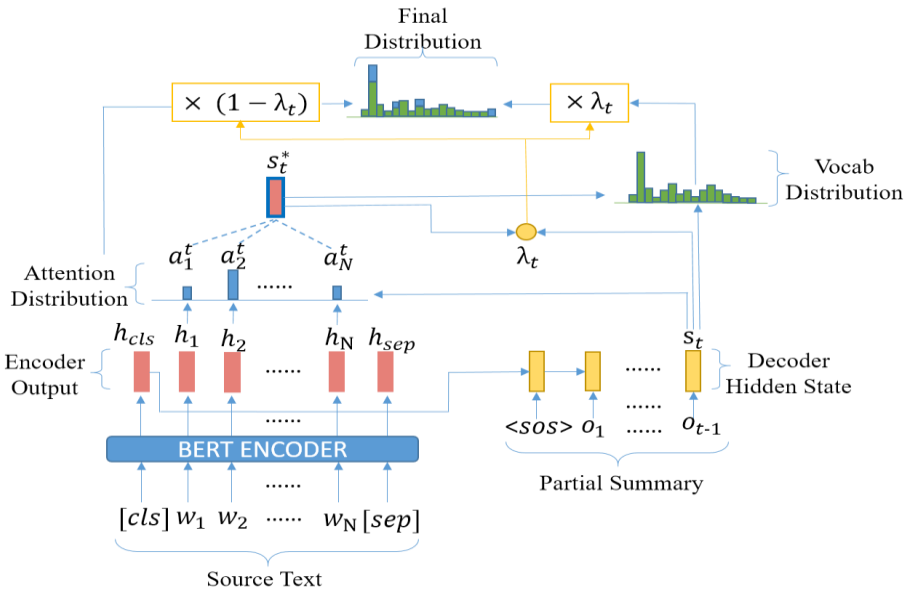
三、方法

本研究嘗試從兩個面向探討現階段重寫式摘要的問題，並藉由指針生成網路、Transformer 和 BERT，提出新穎的重寫式摘要模型，期望增進重寫式摘要的成效。第一個面向是編碼器是否能生成足夠代表文章之表示法：由於解碼器需藉由編碼器生成之文章表示法，來生成出對應此篇文章之摘要，因此如何生成好的文章表示法顯然是一個重要的任務。第二個面向是遞迴神經網路的取代性：當輸入遞迴神經網路的訊息越長，較遠的資訊對於後面時間點的影響越薄弱，但較遠的資訊也可能很重要。此外，遞迴神經網路模型需要依序訓練，無法有效運用平行運算加速處理，常消耗很多時間成本。

有鑑於此，本論文提出兩個新穎的重寫式摘要模型，第一個模型是以指針生成網路為基礎，以 BERT 作為句子表示法的編碼器，期望藉由 BERT 來生成強健且準確的文章表示特徵，使得解碼器能依據此表示法，生成出更好的摘要。第二個模型為第一個模型的延伸，除了使用 BERT 作為編碼器外，更進一步將遞迴神經網路改為使用 Transformer 架構，用注意力機制來獲得時間序列上的關係，藉此解決長時間序列訊息丟失問題，產生更強健且依賴上下文資訊的特徵，期望重寫式摘要的效能再進一步的提升。

（一）以 BERT 為編碼器之指針生成摘要法

為了讓序列對序列模型的編碼器能產生更強健的表示法，我們提出一套以 BERT 為編碼器的指針生成摘要法，其模型架構如圖二所示。我們將文章中文字序列$\{w_1, w_2 \dots, w_n, \dots w_N\}$的前後分別加入$[cls]$與$[sep]$，作為文章開始與結束的符號，輸入到 BERT 編碼器中，BERT 將輸出每個字所對應之向量表示法$\{h_{cls}, h_1, h_2 \dots, h_n, \dots, h_N, h_{sep}\}$，我們將$[cls]$所對應之向量$h_{cls}$視為整篇文章的向量表示法。解碼器端，我們同樣採用傳

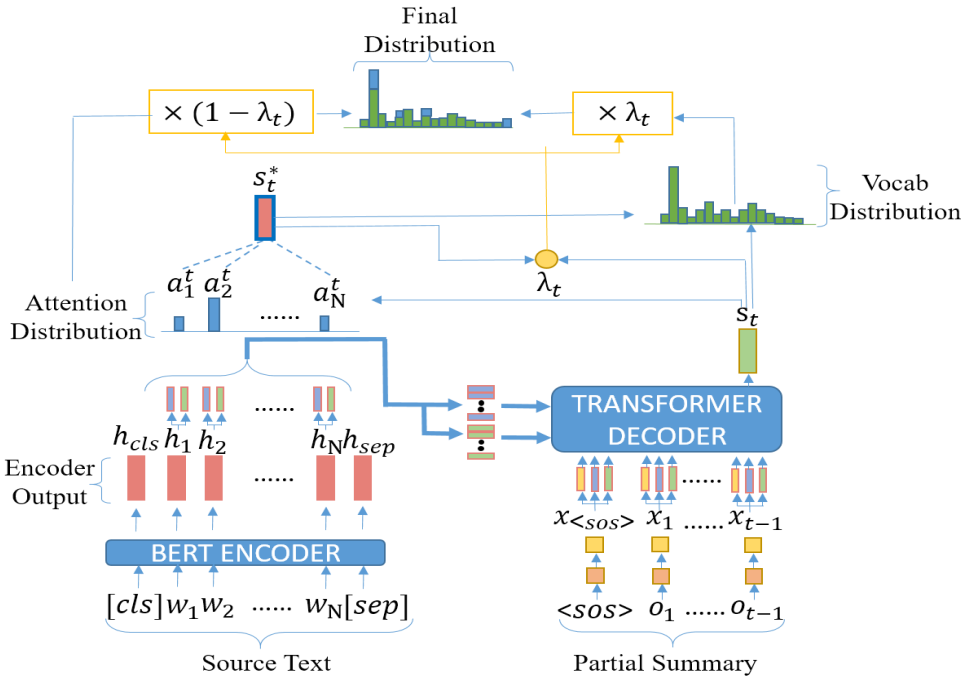


圖二、以 BERT 為編碼器之指針生成摘要法架構圖

統遞迴神經網路為模型基礎。在解碼的過程中，遞迴神經網路會在每個時間點t產生一個輸出s_t，s_t會與編碼器輸出的每個字向量表示法h_n計算出注意力權重$a^t = [a_1^t, \dots, a_n^t, \dots, a_N^t]$，其中$[cls]$與$[sep]$並沒有加入計算。得到注意力權重後，編碼器中每個字向量表示法h_n與時間點t的注意力權重a_n^t相乘並相加後，即得到當前時間點t於解碼器的注意力向量s_t^*。最後，如同指針生成網路，我們使用s_t與s_t^*產生詞彙生成機率分布P_{vocab}；以注意力權重a^t產生P_{PGN}；再利用s_t、s_t^*和解碼器時間點t之輸入x_t計算出P_{vocab}與P_{PGN}的結合係數λ_t，並透過線性組合，產生解碼器在時間點t的參考機率分布$P(w)$（可參閱二-(二)）。與傳統序列對序列、指針生成網路相比，我們改用 BERT 作為文章的特徵抽取器，期待藉由更強健的文章特徵可以產生更佳的重寫式摘要。

（二）融合 BERT 與 Transformer 之指針生成摘要法

此方法為以 BERT 為編碼器之指針生成摘要法之改良模型，方法架構如圖三所示。首先，當我們將文章中文字序列$\{w_1, w_2 \dots, w_n, \dots w_N\}$的前後分別加入$[cls]$與$[sep]$，輸入到 BERT 編碼器之中，獲得對應之向量表示法$\{h_{cls}, h_1, \dots, h_n, \dots, h_N, h_{sep}\}$後，將每個字向量$h_n$各自乘上$K^1, V^1$兩個矩陣，產生兩個對應向量$k_n$與$v_n$，這兩組向量（即$\mathbf{k}_1^N = \{k_1, k_2 \dots, k_n, \dots, k_N\}$與$\mathbf{v}_1^N = \{v_1, v_2 \dots, v_n, \dots, v_N\}$）將用於解碼器之中。在解碼的過程中，當我們解碼第$t$時間點時，會藉由 Transformer 機制，考慮時間點t之前所生成的所有字詞$\{o_1, o_2 \dots, o_{t-1}\}$。更明確地，我們先將$\{o_1, o_2 \dots, o_{t-1}\}$通過詞向量層(Embedding Layer)，



圖三、融合 BERT 與 Transformer 之指針生成摘要法

轉換為詞向量後，$\{h_{o_1}, h_{o_2}, \dots, h_{o_{t-1}}\}$透過$Q^2, K^2, V^2$三個權重矩陣進行轉換，為每一個字分別產生三個向量表示法，即$\mathbf{q}_{o_1}^{o_{t-1}} = \{q_{o_1}, q_{o_2}, \dots, q_{o_{t-1}}\}$、$\mathbf{k}_{o_1}^{o_{t-1}} = \{k_{o_1}, k_{o_2}, \dots, k_{o_{t-1}}\}$與$\mathbf{v}_{o_1}^{o_{t-1}} = \{v_{o_1}, v_{o_2}, \dots, v_{o_{t-1}}\}$。接下來，我們依照 Transformer 自我注意(Self-attention)的機制，計算時間點t時的表示法s_t，最後我們再把編碼器的資訊也考慮進來產生s_t^*:

$$s_t = softmax\left(\frac{q_{o_{t-1}}\left(\boldsymbol{k}_{o_1}^{o_{t-1}}\right)^{\intercal}}{\sqrt{dim}}\right)\boldsymbol{v}_{o_1}^{o_{t-1}} \qquad \text{式(7)}$$

$$s_t^* = softmax\left(\frac{s_t\left(\boldsymbol{k}_1^N\right)^{\intercal}}{\sqrt{dim}}\right)\boldsymbol{v}_1^N \qquad \text{式(8)}$$

dim表示向量的維度，$\intercal$表示矩陣的轉置，並且$softmax\left(\frac{s_t\left(\boldsymbol{k}_1^N\right)^{\intercal}}{\sqrt{dim}}\right)$即為注意力權重$a^t = [a_1^t, \dots, a_n^t, \dots, a_N^t]$。如同指針生成網路，我們使用$s_t$與$s_t^*$產生詞彙生成機率分布$P_{vocab}$；以注意力權重$a^t$產生$P_{PGN}$；再利用$s_t$、$s_t^*$和解碼器時間點$t$之輸入$x_{t-1}$計算出$P_{vocab}$與$P_{PGN}$的結合係數$\lambda_t$，並透過線性組合，產生解碼器在時間點$t$的參考機率分布$P(w)$（可參閱二-(二)）。，此一方法不僅使用 BERT 作為編碼器，更進一步地使用 Transformer 取代傳統的遞迴神經網路，期望這套融合 BERT 與 Transformer 之指針生成摘要法不僅可以平行運算，也可以獲得更好的摘要成效。

表一、基礎系統與本論文所提出方法之實驗結果

		ROUGE-1	ROUGE-2	ROUGE-L
Baseline Systems	seq2seq	0.225	0.153	0.211
	PGN	0.451	0.323	0.368
Our Approaches	Method 1	0.499	**0.346**	0.397
	Method 2	**0.508**	0.340	**0.403**

表二、進階的基礎系統實驗結果

		ROUGE-1	ROUGE-2	ROUGE-L
Advanced Baseline Systems	seq2seq	0.243	0.157	0.212
	PGN	0.469	0.335	0.371

四、實驗設定與結果討論

（一）實驗設定

本論文使用之資料集為 MATBN 中文摘要資料集，共有 205 則文章與對應之摘要，我們依照 80%、10%、10%的比例將資料集隨機切分為訓練集、驗證集以及測試集。實驗結果為三次隨機切分資料集的平均分數。衡量指標為召回率導向的摘要評估(Recall-Oriented Understudy for Gisting Evaluation, ROUGE)[12]，以 ROUGE-1、ROUGE-2 與 ROUGE-L 三種指標為衡量標準。參數設置方面，詞向量設定為 128，隱藏層設定為 256，丟失率設定為 0.3，採用 Adam 優化器，學習率設定為 0.001，批次大小設定為 32。Transformer 參數設置則依照原論文之設置[6]，而層數則設定為 1 層。

（二）實驗結果

首先第一組的實驗中，我們先測試基礎系統（即序列對序列模型(seq2seq)與指針生成網路(PGN)），在中文重寫式摘要的成效，實驗結果如表一所示。實驗結果發現，序列對序列模型在重寫式摘要上已經展現了一定的生成能力。當比較指針生成網路與序列對序列模型時，指針生成網路的 ROUGE 分數有大幅度提升，其中 ROUGE-1 與 ROUGE-2 分數更是兩倍以上。我們認為指針生成網路類似於生成式摘要與抽取式摘要的結合，不僅從文章中抽取重要的詞彙作為摘要，也從詞彙表中選擇合適的詞彙加入摘要，在這種架構下能夠學習語句更通順且更有代表性的摘要，進而提升摘要生成之品質。

接著，我們檢驗本研究所提出的兩個方法，實驗結果如表一所示。首先，我們所提出的第一個方法「以 BERT 為編碼器之指針生成摘要法」(Method 1)，相較於序列對序列模型，成效有大幅度的提升；而當我們進一步比較指針生成網路與我們提出的以 BERT 為編碼器之指針生成摘要法，在 3 種評估指標中，分別有 9.6%、6.8%及 7.1%的相對提升。此結果顯示相較於遞迴神經網路，BERT 能取出更為強健的文章與字詞向量表示法，使得解碼器能透過此表示法生成出更具代表性的文章摘要。而當我們進一步地將遞迴神經網路架構以 Transformer 架構取代，則形成我們所提出的第二個方法「融合 BERT 與 Transformer 之指針生成摘要法」(Method 2)，其實驗結果顯示，這個結合以 BERT 作為特徵抽取、Transformer 作為模型的主幹，再輔以指針生成網路的概念所形成的重寫式摘要法，可以獲得相當優良的實驗成效。

最後，為了分析 Transformer 與傳統遞迴神經網路的差異，我們試著將基礎系統的解碼器皆以 Transformer 取代傳統的長短期記憶模型，其實驗結果如表二所示。比較表二與表一中的基礎系統，實驗結果顯示使用 Transformer 於解碼器的序列對序列模型與指針生成網路，ROUGE 分數皆較使用長短期記憶模型的基礎系統有所提升，說明了 Transformer 確實較傳統遞迴神經網路有較好的效能。

五、結論

本論文透過兩個面向來探討現今重寫式摘要模型，第一個面向是如何在編碼器生成更強健且更代表文章內容之文章表示法，使解碼器端根據更完整的文章訊息，生成出更好的文章摘要。第二個面向是如何解決遞迴神經網路長時間序列下信息丟失問題，使長序列下的文字也能維持著強健的依賴關係。因此，本論文提出兩種新穎的摘要方法，即「以 BERT 為編碼器之指針生成摘要法」與「融合 BERT 與 Transformer 之指針生成摘要法」，並驗證於中文的摘要資料集中。實驗顯示，本研究所提出之改進方法，確實有效地提升摘要任務的成效。在未來，我們會持續深入地探討使用不同 Transformer 層數對於摘要的效能影響外，亦希望能將目前更新穎的 XLNET[13]也應用於重寫式摘要的任務之中。

六、致謝

This work is supported by the Ministry of Science and Technology (MOST) in Taiwan under

grant MOST 108-2636-E-011-005 (Young Scholar Fellowship Program), and by the Project J367B83100 (ITRI) under the sponsorship of the Ministry of Economic Affairs, Taiwan.

參考文獻

[1] A. M. Rush, S. Harvard, S. Chopra, and J. Weston, "A Neural Attention Model for Sentence Summarization," in *ACLWeb. Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2017.

[2] R. Paulus, C. Xiong, and R. Socher, "A deep reinforced model for abstractive summarization," *arXiv:1705.04304,* 2017.

[3] R. Nallapati, B. Zhou, C. Gulcehre, and B. Xiang, "Abstractive text summarization using sequence-to-sequence rnns and beyond," *arXiv preprint arXiv:.06023,* 2016.

[4] J. Gehring, M. Auli, D. Grangier, D. Yarats, and Y. N. Dauphin, "Convolutional sequence to sequence learning," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 2017, pp. 1243-1252: JMLR. org.

[5] S. Chopra, M. Auli, and A. M. Rush, "Abstractive sentence summarization with attentive recurrent neural networks," in *Proceedings of the the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2016, pp. 93-98.

[6] A. Vaswani *et al.*, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998-6008.

[7] I. Sutskever and O. Vinyals, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104-3112.

[8] S. Hochreiter and J. J. N. c. Schmidhuber, "Long short-term memory," vol. 9, no. 8, pp. 1735-1780, 1997.

[9] A. M. Rush, S. Chopra, and J. Weston, "A neural attention model for abstractive sentence summarization," *arXiv:.00685,* 2015.

[10] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv:1810.04805,* 2018.

[11] H. Zheng and M. Lapata, "Sentence Centrality Revisited for Unsupervised Summarization," *arXiv:.03508,* 2019.

[12] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text summarization branches out*, 2004, pp. 74-81.

[13] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, and Q. V. Le, "XLNet: Generalized Autoregressive Pretraining for Language Understanding," *arXiv:1906.08237,* 2019.