The 35th Conference on Computational Linguistics and Speech Processing (ROCLING 2023)
Taipei City, Taiwan, October 20-21, 2023. The Association for Computational Linguistics and Chinese Language Processing

WordRank: A Word Ranking based Training Strategy for Abstractive Document Summarization
(一套基於詞排名的抽象式文件摘要模型訓練法)

Hsiao-Wei Chou[1], Ping-Yen Wu[1], Jia-Jang Tu[2], and Kuan-Yu Chen[1]
[1]National Taiwan University of Science and Technology
[2]Industrial Technology Research Institute
victor88041559@gmail.com brian.92308@gmail.com
santu@itri.org.tw kychen@mail.ntust.edu.tw

摘要

文件摘要一直是個經典且重要的研究議題，旨在將給定的一篇文章濃縮成簡潔精鍊的小段落。關鍵字 (Keyword) 在文章內扮演著承上啓下的重要角色，它們通常乘載著文章的主題、重點與核心概念。於是，過去許多研究提出基於關鍵字的文件摘要模型。然而，這些模型通常由一個萃取關鍵字模型，以及一個將關鍵字做爲導引的摘要生成模型。這樣的設計不僅增加流程的複雜度，可能遭遇錯誤傳遞的問題，也將導致多餘的資源消耗。有鑑於此，本研究致力於提出一套基於詞排名的抽象式摘要模型訓練法，著眼於將關鍵字萃取與文件摘要兩者合而爲一。爲此，模型不僅可以自動地標示出文章內的關鍵字，亦可根據這些關鍵字產生文章的抽象式摘要。實驗結果顯示，使用基於詞排名的模型訓練法後，確實可以有效地提升摘要的成效，並且在關鍵字擷取的任務裡，也可獲得很好的成績。

Abstract

Document summarization has always been a classic and important research topic, aiming to condense a given article into a few concise paragraphs. Keywords, which usually convey the theme, focus, and core concept of the content, play an essential role in the document. Therefore, many studies in the past have proposed keyword-based document summarization models. However, these models usually consist of a keyword extractor and a keyword-based summarizer. Such a design not only increases the complexity of the process but also may encounter an error propagation problem and will also lead to redundant resource consumption. In view of this, this research dedicates to proposing a word ranking based training strategy for abstractive document summarization, which mainly focuses on combining keyword extraction and document summarization. On top of the training strategy, the resulting model can automatically select keywords in the document and generate an abstractive summary based on these keywords. The experimental results show that using the proposed training strategy can indeed effectively improve the quality of the abstractive summarization and achieve good results in the keyword extraction task.

關鍵字：文件摘要、關鍵字、抽象式摘要

1 介紹

文件摘要通常被區分爲抽取式 (Extractive) 與抽象式 (Abstractive) 兩大類。抽取式摘要是從給定的文章中，挑選數個句子來組成摘要；抽象式摘要則是希望摘要像是以閱讀文章後重寫的方式，用自動的方式產生數個句子來做爲摘要。由於自動生成的句子容易有不通順、含有錯字、文法錯誤等問題，因此過去數十年來的摘要研究，多半以抽取式摘要爲主。近年來，深度學習 (Deep Learning) 的蓬勃發展，特別在自然語言處理領域上的屢屢突破，自然語言生成 (Natural Language Generation) 已邁入下一個世代。自動地生成文法正確、沒有錯字且通順的句子，已不再是難以達成的目標。因此，抽象式摘要，成爲近期摘要研究的熱門議題。在基於類神經網路的模型架構下，抽象式摘要任務通常被表示爲一個序列至序列 (Sequence-to-Sequence) 的問題，也就是在給定一篇文章後，模型需要根據這篇文章產生一段文字序列，作爲文章的摘要。

在深度學習的框架下，序列至序列問題的發展，可追朔由遞迴式神經網路 (Recurrent Neural Network, RNN) 開始 (Nallapati et al., 2016)；接著，長短期記憶模型 (Long Short-term Memory, LSTM) 與閘道循環單元模型 (Gated Recurrent Unit, GRU) 等改善遞迴式神經網路的模型紛紛提出 (Li et al., 2018; Shi et al., 2021)；爾後，變形器模型 (Transformer) 不僅改善了各種遞迴式模型計算耗時的問題、利用簡單的自注意力機制 (Self-attention) 來考慮字符 (Token) 與字符之間的

關係，也在各種自然語言處理相關的任務中展現絕佳成效 (Vaswani et al., 2017; Raffel et al., 2020; Reid et al., 2021; Fan et al., 2021)。從此，變形器模型成爲自然語言處理領域中序列至序列問題的主流架構。

基於提示 (Prompt-based) 的摘要模型是近年另一個重要的研究方向，廣義概念是讓模型在產生摘要時，考慮使用者輸入的導引或提示 (Liu and Chen, 2021; Luo et al., 2022; Narayan et al., 2021; Ravaut et al., 2023) 。更明確地，於給定一篇欲摘要的文章後，得以同時將關鍵字或者焦點句子當作提示，一併輸入摘要模型，而模型需要生成以給定的關鍵字或者焦點句子爲主的摘要。延續這個想法，現今許多研究將摘要任務拆解成兩大部分，一個是關鍵字或焦點句子的萃取，一個是結合關鍵字或焦點句子與文章來生成摘要。前者可以看成是對文章進行剖析，將重點預先標註；後者則是根據文章的重點，進行摘要的生成。在這樣的架構下，多數的方法利用串接的方式來達成此一目標，也就是建立一個關鍵字或焦點句子抽取器，以及一個結合關鍵字或焦點句子與文章的文件摘要生成模型。然而，如此複雜的流程，不但可能具有錯誤傳遞 (Error Propagation) 的問題，也因爲需要兩階段執行，而變得耗時。此外，維護兩個模型的穩定與效能，更會提高資源的需求量 (He et al., 2022; Dou et al., 2021)。

有鑑於此，本研究提出一套基於詞排名的新穎性抽象式摘要模型訓練法，我們簡稱爲 WordRank，主要有三大貢獻。第一，爲了解決傳統串接兩個模型之缺點，我們提出的摘要模型訓練法能將關鍵字的抽取與摘要的生成融合爲一，也就是在給定一篇文章後，摘要模型不僅可以自動地標示出關鍵字，也能夠依此產生摘要。第二，這套新穎的抽象式摘要模型訓練法可以用於訓練各式基於變形器架構的抽象式摘要模型，極具彈性與穩定性。最後，我們將這套方法使用於最單純的基於變形器之編碼器-解碼器架構的抽象式摘要模型以及 Pegasus (Zhang et al., 2020) 與 BART (Lewis et al., 2020) 兩個經典的抽象式摘要法。實驗結果顯示，使用本研究提出的訓練法後，各式模型皆能有效地提升抽象式摘要的任務成效。

2 相關研究

2.1 提示學習 (Prompt Learning)

近年來，提示學習已在自然語言處理的領域中被廣泛討論，包含如何有效率地將大型預訓練語言模型運用於各種下游任務 (Liu et al., 2023) 以及各式自然語言生成的相關問題 (Radford et al., 2019; Brown et al., 2020; Schick and Schütze, 2021; Li and Liang, 2021)。對於抽象式摘要任務來說，提供摘要模型額外的指示或導引，使模型可以生成更符合目標或條件的摘要，即是提示學習在摘要上的應用。CtrlSum (He et al., 2022) 提出了一種可控制的摘要框架，利用自動提取的關鍵字和不同的提示來對模型輸出進行 5 種不同方面的控制，使得最終的摘要成果可以有效的提升。GSum (Dou et al., 2021) 是以變形器模型爲架構，在一般常見的文章編碼器外，加入了一個引導資訊編碼器，使得各種不同的導引訊號 (關鍵字、關鍵句等) 能對於摘要的生成產生影響。

2.2 對比學習 (Contrastive Learning)

對比學習 (Hadsell et al., 2006) 已經被廣泛運用在神經網路模型，作爲一種自監督學習的方式。其中，SimCLR (Chen et al., 2020) 將對比學習應用於圖像分類領域，證明了以對比損失 (Contrastive Loss) 訓練的神經網路相較於自監督學習 (Self-supervised) 或半監督學習 (Semi-supervised) 方法，能夠獲得更好的任務成效。很快地，相關研究也被介紹至自然語言處理領域中。SimCSE (Gao et al., 2021) 提出了一個應用對比學習的架構，並針對預訓練語言模型的句嵌入 (Sentence Embedding) 進行訓練的方法，在文本語意相似性 (Semantic Textual Similarity, STS)(Yang et al., 2018) 任務中達到超越無監督學習 (Unsupervised) 和半監督學習的成績。而針對文本生成乃至抽象式摘要任務，SimCLS (Liu and Liu, 2021) 提出了一種將對比學習應用至序列至序列的生成任務框架中，透過比較生成文本之間的品質來建構對比損失，目的是選出最佳的生成文本。更進一步的，BRIO (Liu et al., 2022) 則提出了將傳統文本生成任務中通常使用的交叉熵損失 (Cross Entropy Loss) 與比較文本品質的對比損失相結合，使模型同時擁有生成與評分的能力，而這個訓練方式也使模型在抽象式摘要任務上，獲得非常好的成績。

3 新穎的抽象式摘要訓練法

3.1 基於變形器的基礎抽象式摘要模型

抽象式摘要任務是一個典型的序列至序列的問題，也就是給定一篇欲摘要的文章 $D = \{w_1, ..., w_{|D|}\}$ 後，由機器自動地產生對應的摘要 $Y = \{w_1, ..., w_{|Y|}\}$。以變形器爲基礎時，最基本的模型架構爲將編碼器 (Encoder) 與解碼器 (Decoder) 串接的形式。更明確地，我們首先將文章中的每一個字符 (To-

ken) 轉換成向量表示法，並與位置向量相加，形成一組代表文章裡字符序列的表示法 $H^0_{enc} = \{h^0_1, ...h^0_{|D|}\}$。接著，我們將 H^0_{enc} 輸入進由 L_{enc} 層變形器組成的編碼器，每一個變形器主要由自注意力機制 (Self Attention)、殘差網路 (Residual Network)、層正規化 (Layer Normalization) 以及前饋網路 (Feed-forward Network) 所組成：

$$\begin{aligned} \hat{H}^{l-1}_{enc} &= LN(H^{l-1}_{enc} + SA(H^{l-1}_{enc})) \\ H^{l}_{enc} &= LN(\hat{H}^{l-1}_{enc} + FFN(\hat{H}^{l-1}_{enc})) \end{aligned} \quad (1)$$

其中 SA 代表自注意力機制、LN 表示層正規化、FFN 則爲前饋網路，而 $l \in \{1, 2, ..., L_{enc}\}$ 指的是第幾層變形器。值得一提的是，自注意力機制是注意力機制 (Attention) 的變形 (Lin et al., 2017)，它將輸入透過簡單的前饋網路轉換爲查詢 (Query)、鍵項 (Key) 與值項 (Value)：

$$\begin{aligned} Q^{l-1} &= W^{l-1}_Q H^{l-1}_{enc} \\ K^{l-1} &= W^{l-1}_K H^{l-1}_{enc} \\ V^{l-1} &= W^{l-1}_V H^{l-1}_{enc} \end{aligned} \quad (2)$$

其中，$\{W^{l-1}_Q, W^{l-1}_K, W^{l-1}_V\}$ 爲前饋網路的模型參數，$\{Q^{l-1}, K^{l-1}, V^{l-1}\}$ 分別表示查詢、鍵項與值項。接著，利用查詢與鍵項計算每一個字符與所有字符的相關係數：

$$SA(H^{l-1}) = softmax(\frac{Q^{l-1}{K^{l-1}}^T}{\sqrt{d_{model}}})V^{l-1} \quad (3)$$

其中 $\sqrt{d_{model}}$ 爲一個縮放係數 (Scaling Factor)，d_{model} 則是向量的維度。最後，將相關係數與對應的值項相乘，加總後即成爲每一個字符新的向量表示法 (Bahdanau et al., 2015; Luong et al., 2015)。

在編碼器對文章中每一個字符都產生一個對應的向量後，抽象式摘要的生成，是藉由解碼器來完成。解碼器同樣由 L_{dec} 個堆疊的變形器所組成，除了自注意力機制、殘差網路、層正規化以及前饋網路外，解碼器中的變形器還包含有交叉注意力機制 (Cross Attention)。更明確地，基於變形器的抽象式摘要法是以循序的方式，一個字一個字依序的輸出，所以當要產生第 y 個字符 w_y 時，解碼器的輸入爲 $Y_{<y} = \{w_1, w_2, ..., w_{y-1}\}$。同樣地，我們將每一個字符轉換成向量表示法，並與位置向量相加，形成 $H^0_{dec} = \{h^0_1, ...h^0_{y-1}\}$，再輸入進解碼器中進行運算：

$$\begin{aligned} \hat{H}^{l-1}_{dec} &= LN(H^{l-1}_{dec} + SA(H^{l-1}_{dec})) \\ \bar{H}^{l-1}_{dec} &= LN(\hat{H}^{l-1}_{dec} + CA(H^{L_{enc}}_{enc}, \hat{H}^{l-1}_{dec})) \\ H^{l}_{dec} &= LN(\bar{H}^{l-1}_{dec} + FFN(\bar{H}^{l-1}_{dec})) \end{aligned} \quad (4)$$

其中 CA 爲交叉注意力機制。交叉注意力機制與自注意力機制的運算方式完全相同（參考式 2 與 3），差異僅在交叉注意力機制使用 $\hat{H}^{l-1}_{dec}$ 產生運算時所需的查詢，而利用 $H^{L_{enc}}_{enc}$ 產生鍵項與值項 (Vaswani et al., 2017)。這個設計，使得模型在生成摘要時，不但可以基於已經生成的字符序列資訊（即 $\hat{H}^{l-1}_{dec}$），也能夠同時參考文章的資訊（即 $H^{L_{enc}}_{enc}$）。最後，我們使用負對數相似度作爲抽象式摘要模型參數訓練時的損失函數：

$$\mathcal{L}_{MLE} = -\sum_{y=1}^{|Y|} \log P(w_y | w_{<y}, D) \quad (5)$$

3.2 基於詞排名的抽象式摘要模型訓練法

在基於變形器的抽象式摘要模型架構下，自注意力機制與交叉注意力機制在編碼器與解碼器內扮演重要的角色。在編碼器中，自注意力機制讓文章內的字符透過兩兩交互的比對計算，總結出每一個字符的高層次語意向量表示法；在解碼器內，自注意力機制探索著已經被解碼出的字符序列內的語意內容，再搭配交叉注意力機制，將文章的資訊與已被解碼的字符序列一併考慮，決定接下來要再生成的摘要內容。更進一步地，在交叉注意力機制的運算中，字符與字符之間的關係，是以內積的方式進行（參考式 3），因此字符向量表示法內每一個維度”值”的大小，會直接地影響模型對某些字符的關注度。也就是說，字符向量表示法的長度越長（即每個維度的值較大），越可能讓模型聚焦關注這個字符。因此，若能讓文章中關鍵字的向量表示法長度較長，模型在生成摘要時，就能讓模型自動地著重這些關鍵字，進而生成品質更好的抽象式摘要結果。

有鑑於此，本研究提出一套基於詞排名的抽象式摘要模型訓練法，期望摘要模型可以自動地關注文章中可能的關鍵字，並以這些關鍵字作爲提示，生成品質更好、內容更精準的抽象式摘要。爲達此一目的，我們首先對訓練資料集 $\mathcal{D} = \{(D_1, Y_1), ..., (D_{|\mathcal{D}|}, Y_{|\mathcal{D}|})\}$ 中的每一個摘要答案 Y 進行詞性標註 (Part-of-speech Tagging)；接著，我們去掉詞性爲連接詞 (CC)、數字 (CD)、樣態輔助詞 (MD)、限定詞 (DT)、介詞 (IN)、第三人稱單數現

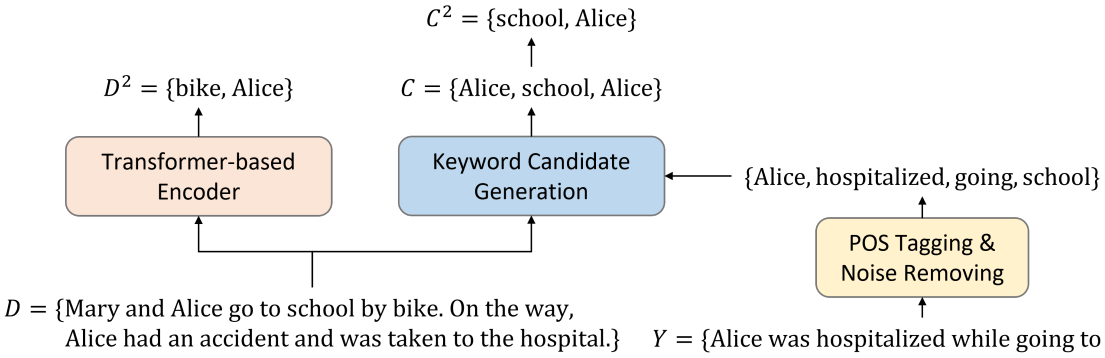

Figure 1: 基於詞排名的抽象式摘要模型訓練法之流程示意圖。以文章 D = {Mary and Alice go to school by bike. On the way, Alice had an accident and was taken to the hospital.} 和摘要 Y = {Alice was hospitalized while going to school.} 爲例。在對摘要答案 Y 進行詞性標註與過濾後，留下了 {Alice, hospitalized, going, school} 四個字，因爲 Alice 與 school 有在文章中出現，所以這篇文章中的候選關鍵字依序爲 C = {Alice, school, Alice}，即 $N = 3$；接著，若我們選取長度最長的 2 個字符，也就是將 K 設爲 2，可以得到 C^2 = {school, Alice} 和 D^2 = {bike, Alice}，也可以計算他們的差集 $C^2 \setminus D^2$ = {school}，而 $D^2 \setminus C^2$ = {bike}。

在式動詞 (VBZ)，以及英文中的特殊介詞 to(TO)、wh 開頭之副詞 (WRB)、wh 開頭之代名詞 (WP)、wh 開頭之定冠詞 (WDT)、wh 開頭之所有格 (WP$)，並且將停用詞（Stop Word）也一併濾除；最後，剩下的字符如果有在對應的文章 D 中出現，我們則將文章內這些字符視爲候選關鍵字 $C = \{w_1^C, ..., w_N^C\}$。值得注意的是，候選關鍵字個數 N 必將小於等於文章 D 的長度 $|D|$，並且 C 內可能出現重複的字符。相關過程如圖1所示。

文章 D 在經過編碼器的處理後，每一個字符有其對應的向量表示法 $E = \{e_{w_1}, ..., e_{w_{|D|}}\}$，因此我們可以爲每一個字符計算其向量表示法長度，也可獲得每一個候選關鍵字的向量長度。根據向量表示法的長度，我們在候選關鍵字中選取前 K 大的字符 $C^K = \{w_1^{C^K}, ..., w_K^{C^K}\}$，也在文章中挑選向量表示法長度最長的前 K 個字符 $D^K = \{w_1^{D^K}, ..., w_K^{D^K}\}$。$D^K$ 表示當前模型，在交叉注意力機制下將會較爲關注的前 K 個字符，而 C^K 則爲我們認爲交叉注意力機制較應關注的前 K 個關鍵字。換句話說，我們希望 C^K 與 D^K 盡量全等，也就是希望模型在產生抽象式摘要時，參考的關鍵字與我們所選定的候選關鍵字應該一致。因此，訓練時的損失函數定義爲：

$$\mathcal{L}_{WR} = max(0, \hat{S} - \bar{S} + margin) \quad (6)$$

$$\hat{S} = \sum_{w_i \in D^K \setminus C^K} \frac{|e_{w_i}|}{|D^K \setminus C^K|} \quad (7)$$

$$\bar{S} = \sum_{w_i \in C^K \setminus D^K} \frac{|e_{w_i}|}{|C^K \setminus D^K|} \quad (8)$$

$$margin = |D^K \setminus C^K| * \frac{|D|}{N * \sqrt{d_{model}}} \quad (9)$$

其中，$D^K \setminus C^K$ 代表集合 D^K 與 C^K 的差集，$|D^K \setminus C^K|$ 表示差集內的元素個數，$|e_{w_i}|$ 表示字符向量 e_{w_i} 的長度。值得一提的是，$|D^K \setminus C^K|$ 與 $|C^K \setminus D^K|$ 並定是相等的；$margin$ 代表一個容忍值，其值的大小取決於差集的大小、文章中候選關鍵字數 N、選取的關鍵字數 K 以及字符的向量表示法維度 d_{model}。最終，我們結合傳統抽象式摘要模型的訓練目標與基於詞排名的損失函數，作爲模型參數更新的依歸：

$$\mathcal{L} = \mathcal{L}_{MLE} + \mathcal{L}_{WR} \quad (10)$$

在測試階段，當給定一篇欲摘要的文章後，編碼器會爲文章內每一個字符生成一個向量表示法，我們可以藉由計算向量表示法的長度，獲得文章中可能的關鍵字。藉由解碼器，摘要模型能爲這篇文章產生基於關鍵字提示的抽象式摘要。因此，在測試階段，我們不須對文章進行詞性標注，模型將自動地爲可能的關鍵字產生長度較長的向量表示法，並藉此讓摘要在生成時，可以偏重這些可能的關鍵字。最後，值得強調的是，這套基於詞排名的抽象式摘要模型訓練法可與現有各式基於變形器的抽象式摘要模型相結合，極具延展性。

4 實驗

4.1 實驗設定

4.1.1 資料集

在本研究中，我們使用 CNN / Daily Mail News Summarization Dataset (CNNDM[1])

[1]https://cs.nyu.edu/~kcho/DMQA/

	CNNDM			XSUM		
	R-1	R-2	R-L	R-1	R-2	R-L
Naïve Transformer	40.88	17.88	37.80	28.38	9.20	22.60
Naïve Transformer + WordRank	**41.09**	**18.16**	**38.02**	**29.47**	**10.03**	**23.54**
PEGASUS (Zhang et al., 2020)	44.17	21.47	41.11	47.21	24.56	39.25
PEGASUS our	44.20	21.35	41.03	47.21	24.36	39.01
PEGASUS + WordRank	**44.28**	**21.48**	**41.14**	**47.34**	**24.49**	**39.18**
BART (Lewis et al., 2020)	44.16	21.28	40.90	45.14	22.27	37.25
BART our	44.29	21.26	41.05	45.08	22.07	36.83
BART + WordRank	**44.55**	**21.55**	**41.38**	**45.25**	**22.24**	**37.07**

Table 1: PEGASUS 與 BART 在 CNNDM 與 XSUM 資料集的抽象式摘要實驗結果。

(Hermann et al., 2015) 與 Extreme Summarization Dataset (XSUM[2]) (Narayan et al., 2018) 來驗證我們所提出的基於關鍵字提示之抽象式摘要模型訓練法。CNNDM 是由有線電視新聞網（Cable News Network, CNN）和每日郵報（Daily Mail）的新聞文章所組成，我們按造傳統的做法，將新聞文章視爲摘要文件，而亮點 (Highlight) 則做爲該篇文章的抽象式摘要解答 (Nallapati et al., 2016)。XSUM 是以英國廣播公司（British Broadcasting Corporation, BBC）的新聞文章所做成的資料集，相較於 CNNDM，XSUM 的摘要解答通常僅有一句，並且與文章的用字遣詞差異較大，是屬" 高度抽象" 的摘要資料集。

本研究採用 ROUGE（Recall-Oriented Understudy for Gisting Evaluation）(Lin, 2004) 作爲衡量文件摘要效果的評估方法。ROUGE 是一種常用於自然語言處理領域的評估方法，特別用於衡量自動摘要系統生成的摘要與人工撰寫的參考摘要之間的相似度和品質。ROUGE 分數是藉由計算生成的摘要與參考摘要之間重疊的單位元素（例如字母、單詞、詞組）來量化他們的相似程度。其基本概念是，一個好的自動摘要應該涵蓋參考摘要中的關鍵訊息，並且在使用不同的單位元素進行比較時，皆能保持一定的相似性。由於 ROUGE 採用了單位元素比對的方式，避免了涉及語句邊界定義的問題，因此在文件摘要任務中具有廣泛的適用性。特別是在多份摘要結果需要評估的情況下，ROUGE 能夠有效且快速地提供客觀的評價依據。在本研究中，我們採用了 ROUGE-1（Unigram, R-1）、ROUGE-2（Bigram, R-2）以及 ROUGE-L（Longest Common Subsequence, R-L）這三種常用的指標。ROUGE-1 用於衡量自動生成摘要的資訊量，ROUGE-2 則關注於評估摘要的流暢性，而 ROUGE-L 則著重考慮最長的共同子序列。藉由綜合考慮這些指標，我們能夠更全面地評估自動摘要系統在資訊涵蓋、語法連貫性以及核心內容保持等方面的性能表現。

4.1.2 模型架構

由於基於詞排名的抽象式摘要模型訓練法可以與各式基於變形器的摘要模型相結合，因此我們選擇當前極具代表性的 PEGASUS(Zhang et al., 2020) 與 BART(Lewis et al., 2020)，兩種不同的預訓練抽象式摘要模型爲基礎，以及完全初始化參數的 Transformer(Vaswani et al., 2017)，Transformer 的層數、輸入長度以及其餘設定與 BART 模型一致，我們比較加入本研究提出的訓練法後，是否可以有效地增進摘要的成果，以驗證我們方法的有效性。預訓練的 PEGASUS[3]與 BART[4]都是源自於 Transformers Library(Wolf et al., 2020)。在訓練時，優化器爲 Adam (Kingma and Ba, 2015)，而計算 $\mathcal{L}_{MLE}$ 時，會先採用 label smoothing (Szegedy et al., 2016) 的技術來軟化目標分布再進行運算，軟化係數設定爲 0.1，學習率設定爲 $2\times10^{-3}\min(\text{step}^{-0.5}, \text{step}\cdot\text{warmup}^{-1.5})$，其中的熱身步驟 warmup 設定爲 500，step 表示更新步數。在基於詞排名的抽象式摘要模型訓練法中，我們將目標關鍵字個數 K 設定爲所有由文章中挑選出來的候選關鍵字，即 $K = N$(參考章節 3.2)，因此 K 會是一個變數，不是一個固定值。實驗所採用的圖形運算單元 (GPU) 爲 1 張 NVIDIA GeForce RTX 3090；在 CNNDM 資料集上，訓練 1 次迭代 (Epoch) 約需 12 小時，而在 XSUM 資料裡，1 次迭代約需 7 小時，我們在 CNNDM 中運行 2 個迭代，XSUM 則爲 5 個，皆用預訓練後的參數進行比較，而 Transformer 在 CNNDM 中運行 14 個迭代，XSUM

[2]https://github.com/EdinburghNLP/XSum

[3]https://google/pegasus-xsum and https://sshleifer/pegasus-cnn-ft-v2

[4]https://facebook/bart-large-cnn and https://facebook/bart-large-xsum

則爲 23 個。

4.2 實驗結果

4.2.1 抽象式文件摘要

在第一組實驗裡，我們首先比較 PEGASUS 與 BART 模型於抽象式摘要的基礎結果，相關結果如表1所示。除了我們重現的 PEGASUS 與 BART 於摘要任務的成果外，原始論文的相關數據亦呈現於表1中，我們可以發現，本研究的重現成果與原始成績均在伯仲之間，這顯示我們的基礎系統是可靠且合理的。在這些基礎摘要模型之上，我們加入了本研究提出之基於詞排名的抽象式摘要模型訓練法 (WordRank)，相關結果同樣展示於表1中。根據實驗結果，藉由這套訓練法，不論是 PEGASUS 或 BART 模型，都能在抽象式摘要任務上取得亮眼的進步。這個結果顯示，本研究提出的抽象式摘要模型訓練法，不僅確實能夠在摘要產生時，提供給解碼器額外的關鍵字提示，使得最終的摘要結果更好，也展示了這個訓練方法，確實能夠與不同的抽象式摘要模型相結合，並取得更進步的成績！

4.2.2 關鍵字預測

接著，我們進一步地探究模型是否具備自動標示出文章內關鍵字的能力。在這組實驗中，我們以 CNNDM 內的測試集爲例，利用文章所對應的摘要答案，透過詞性標註與過濾，爲每一篇文章標示出一組候選關鍵字（詳細作法請參閱章節 3.2）。接著，我們將文章輸入摘要模型的編碼器，以獲得到每一個字符的向量表示法。藉由向量表示法的長度，即能挑選出文章中可能的關鍵字！因此，在實驗裡，我們計算向量長度最長的前 3、5、10 個字符中，有多少比例是屬於候選關鍵字 (即 *Precision*@3、@5 與 @10)，來評估摘要模型對於關鍵字預測的準確性，相關實驗結果如表2所示。首先，根據實驗結果可以發現，在利用基於詞排名的抽象式摘要模型訓練法後，PEGASUS 與 BART 模型皆能在關鍵字預測的實驗裡有大幅度的精準度提升。此一結果不僅說明本研究提出之訓練法確實可以做爲關鍵字預測之用，同時也說明了因爲關鍵字所對應的字符向量確實被加長了，因此抽象式摘要模型中的交叉注意力機制更能關注在這些可能的關鍵字上，使得摘要的成果如同期待地提升了！

此外，我們亦將這些結果與基於 BERT 的關鍵字預測模型相比較 (Gehrmann et al., 2018; He et al., 2022)，相關結果同樣呈現於表2。BERT_{base} 與 BERT_{large} 分別代表使用 12 與 24 層的變形器模型之大型預訓練語言模型。在預訓練模型之上，我們先使用訓練集進行關鍵字預測的下游任務微調 (Finetune)，訓練目標是爲每一個輸入的字符進行二元分類，判斷是否爲關鍵字。由實驗結果可以發現，各式使用基於詞排名的抽象式摘要模型訓練法訓練而得的模型 (即 PEGASUS+WordRank 與 BART+WordRank)，其關鍵字預測的成績與單純的關鍵字預測模型不相上下，甚至 PEGASUS+WordRank 的任務成效超越了參數量近乎是其兩倍的 BERT_{large}。這組實驗結果令人十分驚艷，因爲本研究提出之基於詞排名的抽象式摘要模型訓練法，確實可以同時將關鍵字預測與抽象式摘要融合爲一，並且在這兩個任務上皆能獲得很好的成果。

	top3	top5	top10
BERT_{base}	72.76%	66.49%	56.43%
BERT_{large}	73.30%	67.70%	57.86%
Naïve Transformer	26.48%	25.91%	22.27%
BART	9.84%	9.76%	9.77%
PEGASUS	11.67%	11.60%	11.37%
Naïve Transformer + WordRank	62.37%	56.77%	45.82%
BART + WordRank	73.80%	67.09%	56.48%
PEGASUS + WordRank	75.22%	68.60%	57.55%

Table 2: 關鍵字預測之實驗結果。

	top3	top5	top10
epoch5	33.81%	32.54%	29.96%
epoch8	32.15%	31.18%	28.85%
epoch11	28.83%	28.27%	26.58%
epoch14	26.48%	25.91%	22.27%

Table 3: Naïve Transformer 迭代的關鍵字預測結果比較。

4.2.3 Naïve Transformer 關鍵字預測結果分析

在表2中可以看出，Naïve Transformer 的 top K 預測結果比 BART , PEGASUS 兩預訓練模型還要高，我們認爲是因爲模型在訓練前後期的行爲不一樣導致的，我們爲此做進一步的分析，如表3，我們打印不同迭代中 Naïve Transformer 的 top K 結果，可以看出模型在迭代越往前的分數反而更高，越訓練後的分數反而越低，而這樣的結果也類似於 (Goyal et al., 2022) 所說的，在微調階段，模型在訓練前期學習的分布會比較平均，但隨著訓練時間增加，參數會越來越去擬和那些簡單的詞彙，如暫停詞或是一些常用的詞彙。當模型越看重這些詞彙，其代表向量長度就越長，進而導致 Naïve Transformer 的 top k 比兩預訓練模型高，我們也認爲在更長時間的練後，當字詞的擬和收斂時，其分數會接近兩預訓練模型的結果。

	CASED			UNCASED		
	R1	R2	RL	R1	R2	RL
Naïve Transformer	40.88	17.88	37.47	40.12	16.94	36.54
Naïve Transformer + WordRank	**41.09**	**18.16**	**38.02**	**40.48**	**17.45**	**37.00**
PEGASUS	44.20	21.35	41.03	43.81	21.06	40.41
PEGASUS + WordRank	**44.28**	**21.48**	**41.14**	**43.94**	**21.18**	**40.54**
BART	44.29	21.26	41.05	44.34	21.33	40.70
BART + WordRank	**44.55**	**21.55**	**41.38**	**44.50**	**21.49**	**41.33**

Table 4: 大小寫資訊對摘要任務的影響之實驗結果以 CNNDM 爲例。

4.2.4 大小寫資訊的影響

在英文文章內，字首大寫的單字通常有其特別之處，像是專有名詞、稱謂等等，而這些特殊詞彙通常在文章中扮演著重要的角色。因此，在這組實驗中，我們將探討英文文件中大小寫對於抽象式摘要任務的成效影響。同樣以 CNNDM 資料集爲例，將文章內所有字母轉成小寫，之後再對模型進行訓練與測試，相關實驗如表4所示。實驗結果與我們的想像十分接近，將文章中的字母全部轉成小寫後 (UNCASED)，一些重要的資訊因此被抹除了，所以與使用原始文章 (即保留著大寫字母, CASED) 的結果相比，多數的實驗成績皆呈現下降的情況。值得一提的是，雖然沒有大寫文字的資訊，但若使用本研究提出的摘要模型訓練法，仍舊可以取得不小的進步。

4.2.5 字符向量長度之變化

在最後一組實驗裡，我們隨機從測試資料集中挑選一篇文章，並以 BART 與 BART+WordRank 模型爲例，比較文章內每一個字符在使用基於詞排名的摘要模型訓練法前後，字符向量表示法長度的變化，相關結果如圖2所示。由結果可知，候選關鍵字在 BART+WordRank 模型中，似乎都能有長度較長的向量表示法，在 BART 模型裡，雖然字符與字符間，向量表示法的長度差異似乎較爲明顯，但也可發現許多候選關鍵字的長度是相對較短的。因此，這組實驗驗證了基於詞排名的摘要模型訓練法，可以盡可能地讓關鍵字的向量表示法長度變長，進而影響最終的摘要任務成效。

5 結論

本研究提出一套基於詞排名的抽象式文件摘要模型訓練法，旨於將關鍵字預測與抽象式文件摘要融合爲一，期望在關鍵字的引導下，能讓抽象式文件摘要內容更精準。此外，這套訓練法可以與各式基於變形器的抽象式摘要模型相結合，極具彈性與泛化能力。一系列的實驗顯示，本研究提出的方法，在抽象式摘要與關鍵字預測的任務中，皆能展現很好的成果。在未來，我們將持續精進此一摘要模型訓練法，並將其概念與方法運用於其他自然語言處理相關的任務之中，諸如抽取式文件摘要、資訊檢索與機器翻譯等。

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.

Zi-Yi Dou, Pengfei Liu, Hiroaki Hayashi, Zhengbao Jiang, and Graham Neubig. 2021. GSum: A general framework for guided neural abstractive summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4830–4842, Online. Association for Computational Linguistics.

Zhihao Fan, Yeyun Gong, Dayiheng Liu, Zhongyu Wei, Siyuan Wang, Jian Jiao, Nan Duan, Ruofei Zhang, and Xuanjing Huang. 2021. Mask attention networks: Rethinking and strengthen transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1692–1701, Online. Association for Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Sebastian Gehrmann, Yuntian Deng, and Alexander Rush. 2018. Bottom-up abstractive summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4098–4109, Brussels, Belgium. Association for Computational Linguistics.

Tanya Goyal, Jiacheng Xu, Junyi Jessy Li, and Greg Durrett. 2022. Training dynamics for text summarization models.

Raia Hadsell, Sumit Chopra, and Yann LeCun. 2006. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE.

Junxian He, Wojciech Kryscinski, Bryan McCann, Nazneen Rajani, and Caiming Xiong. 2022. CTRLsum: Towards generic controllable text summarization. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5879–5915, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. *Advances in neural information processing systems*, 28.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Wei Li, Xinyan Xiao, Yajuan Lyu, and Yuanzhuo Wang. 2018. Improving neural abstractive document summarization with explicit information selection modeling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1787–1796, Brussels, Belgium. Association for Computational Linguistics.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. A STRUCTURED SELF-ATTENTIVE SENTENCE EMBEDDING. In *International Conference on Learning Representations*.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35.

Yixin Liu and Pengfei Liu. 2021. SimCLS: A simple framework for contrastive learning of abstractive summarization. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 1065–1072, Online. Association for Computational Linguistics.

Yixin Liu, Pengfei Liu, Dragomir Radev, and Graham Neubig. 2022. BRIO: Bringing order to abstractive summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2890–2903, Dublin, Ireland. Association for Computational Linguistics.

Zhengyuan Liu and Nancy Chen. 2021. Controllable neural dialogue summarization with personal named entity planning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 92–106, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Zheheng Luo, Qianqian Xie, and Sophia Ananiadou. 2022. Readability controllable biomedical document summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4667–4680, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to

attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal. Association for Computational Linguistics.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar Gülçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290, Berlin, Germany. Association for Computational Linguistics.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.

Shashi Narayan, Yao Zhao, Joshua Maynez, Gonçalo Simões, Vitaly Nikolaev, and Ryan McDonald. 2021. Planning with learned entity prompts for abstractive summarization. *Transactions of the Association for Computational Linguistics*, 9:1475–1492.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1).

Mathieu Ravaut, Hailin Chen, Ruochen Zhao, Chengwei Qin, Shafiq Joty, and Nancy Chen. 2023. Promptsum: Parameter-efficient controllable abstractive summarization. *arXiv preprint arXiv:2308.03117*.

Machel Reid, Edison Marrese-Taylor, and Yutaka Matsuo. 2021. Subformer: Exploring weight sharing for parameter efficiency in generative transformers. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4081–4090, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Timo Schick and Hinrich Schütze. 2021. Few-shot text generation with natural language instructions. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 390–402.

Tian Shi, Yaser Keneshloo, Naren Ramakrishnan, and Chandan K Reddy. 2021. Neural abstractive text summarization with sequence-to-sequence models. *ACM Transactions on Data Science*, 2(1):1–37.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2818–2826.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Yinfei Yang, Steve Yuan, Daniel Cer, Sheng-yi Kong, Noah Constant, Petr Pilar, Heming Ge, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Learning semantic textual similarity from conversations. In *Proceedings of the Third Workshop on Representation Learning for NLP*, pages 164–174, Melbourne, Australia. Association for Computational Linguistics.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR.

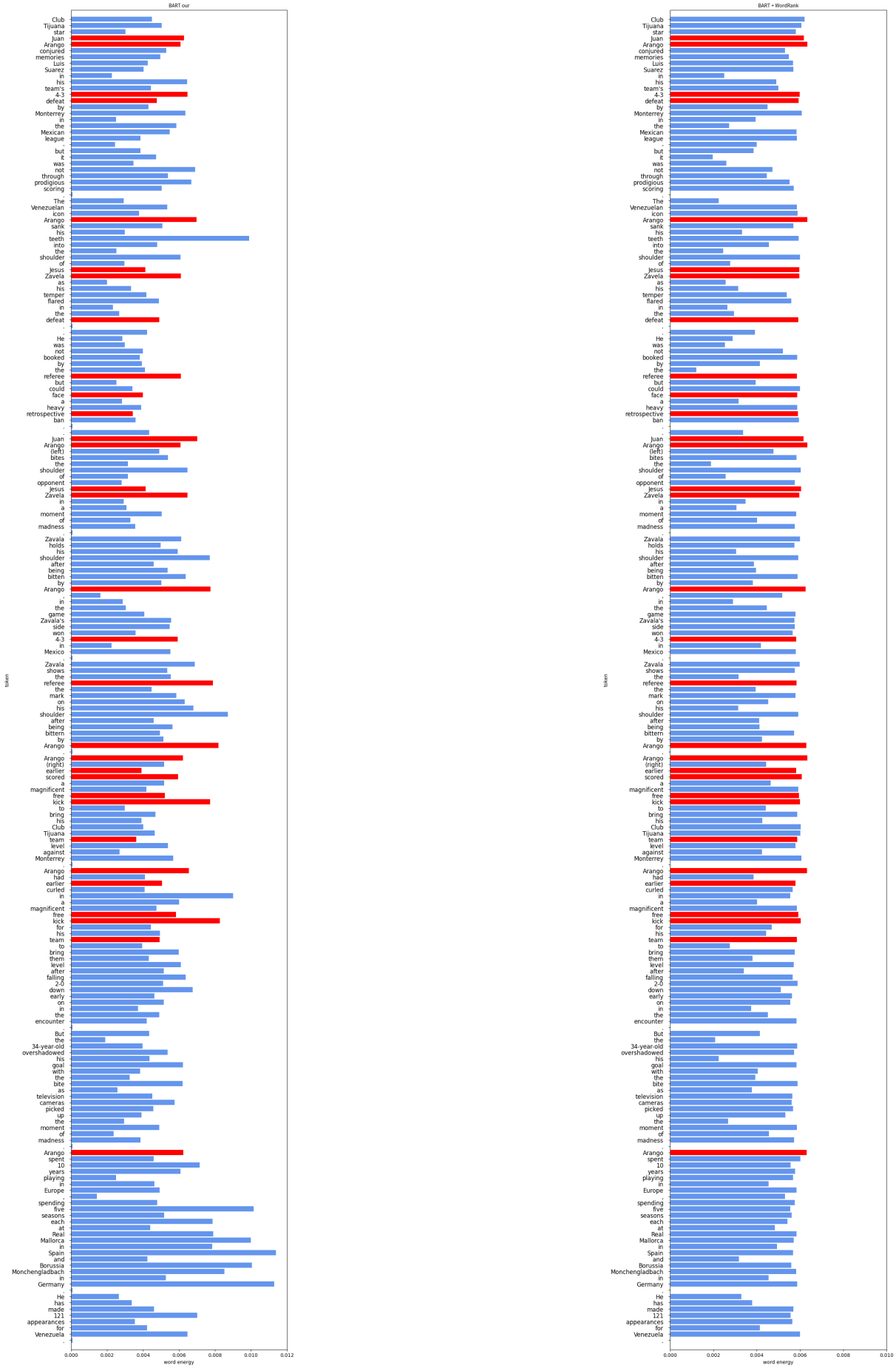

Figure 2: 文章內每一個字符之向量表示法長度圖。左圖爲使用 BART 模型之結果，右圖則爲使用 BART+WordRank 模型之結果，紅色代表該字符爲候選關鍵字。