Machine Learning Final Project Report

News Retrieval – AI CUP 2019 Competition

組隊資訊

NTU_B04901147_B04 戰隊

B04901147 黃健祐 B04901166 陳培鳴

B04104040 解正安 R07943176 高禎謙

題目簡介與動機

Information Retrieval (IR) 是 NLP 中相當經典的題目，討論相關方法的論文也不計其數。在本次比賽中，除了檢索出與 query 相關的新聞外，檢索的結果也必須符合由 query 給定的特定立場，以應用在具爭議性議題的新聞。若能從從大量的新聞文件裡，快速搜尋各種爭議性議題中具特定立場的新聞，除了有助於民眾理解不同角度的觀點，也能使得決策的制定更為全面。

資料處理

我們將新聞文章中的標點符號移除後，使用 jieba 套件進行斷詞。本次實作中我們沒有使用停用詞。另外，我們建立了一個 id to content 的字典，方便以編號查詢新聞。

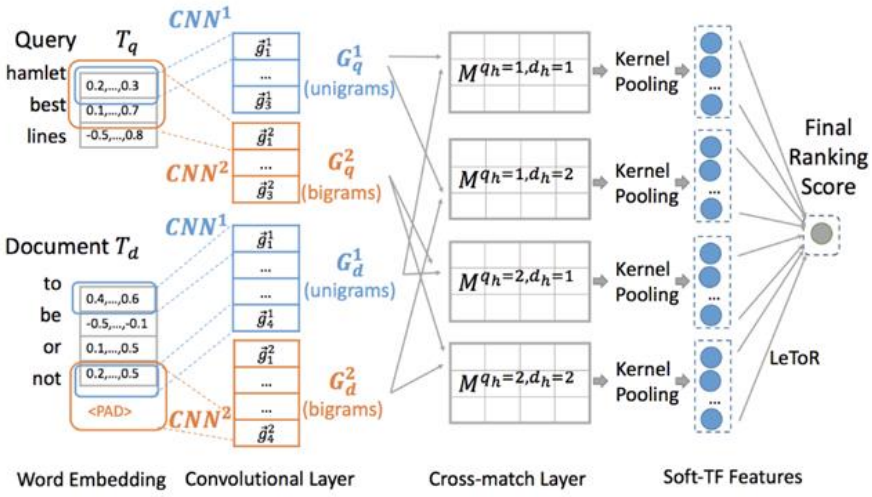
實作方法

本次使用到的方法主要可分為以下兩大類：

Neural Network

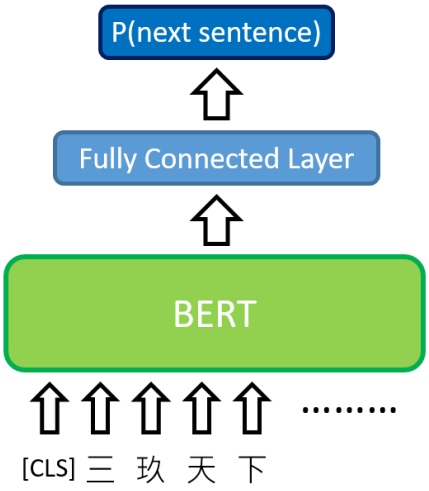
1. Conv-Knrm [1][2]

   Conv-Knrm 是一種利用 convolution 來做 document ranking 的方法，架構如下圖所示。首先將 query 和 context 映射成 word embedding（本次專題使用 dim=256）。接著計算 word vector 的卷積，用來獲得關於鄰近詞的資訊。之後再通過一個 cross-matching layer，計算向量之間的 cosine similarity。最後再通過 RBF kernel 以及 Soft-TF 去計算相似度。我們把訓練分成兩個部分，第一部分是學習如何排序，第二部分則是決定文件和 query 的相似度。結果在兩個部分的正確率都非常低，可能原因是訓練資料太少（不到 5000 筆），訓練資料的組成又特別複雜。



2. BERT [3][4]

   BERT 在 information retrieval 的應用大致與 next sentence prediction 相似：以 query 作為第一個句子，context 當作第二個句子，將兩者合併輸入 BERT 後再接上一層 fully-connected layer，輸出匹配分數（如下圖）。然而，BERT 對於中文的 word segmentation 是以字（character）為單位，而 pre-trained BERT 又有字數限制（512 個字），一般的新聞文章字數大多是超過限制，使得不少語句必須被捨棄。BERT 的 training 相當費時，loss 無法穩定下降，最後輸出的 ranking 結果也相當不理想，因此最後決定捨棄使用。



Traditional Method

3. TF-IDF [5]

   TF-IDF 是排序關鍵詞重要性的指標，由 TF（term frequency）以及 IDF（inverse document frequency）兩項指標組成。其中：

$$TF_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

$$IDF_i = log\left(\frac{|D|}{|\{j: t_i \in d_j\}|}\right)$$

   TF 指的是一個關鍵詞在某文件中的詞頻，IDF 則是出現過某關鍵詞的文件數的倒數。當一個關鍵詞在某個文件中有很高的詞頻（高 TF），而且只出現在少數文件（高 IDF）時，它就會被認為對文章的分類有重要的影響。

4. BM25 [6]

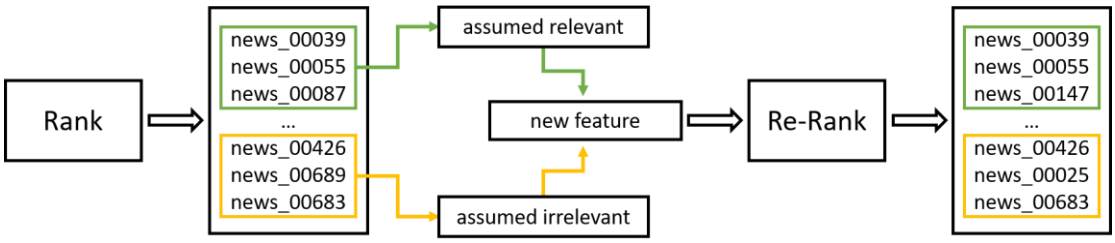
   BM25 是 TF-IDF 的改良版本：

$$score(D,Q) = \sum_{i=1}^{N} IDF(q_i) \frac{TF(q_i,D)(k+1)}{TF(q_i,D) + k\left(1 - b + b\frac{|D|}{avgdl}\right)}$$

   在 TF-IDF 中，一個詞的重要性可以任意的大，然而實際上任何一個詞都不可能對一份文件有無限大的重要性，因此在 BM25 中就把詞彙的分數上限訂為一個參數，來彌補 TF-IDF 的不足。

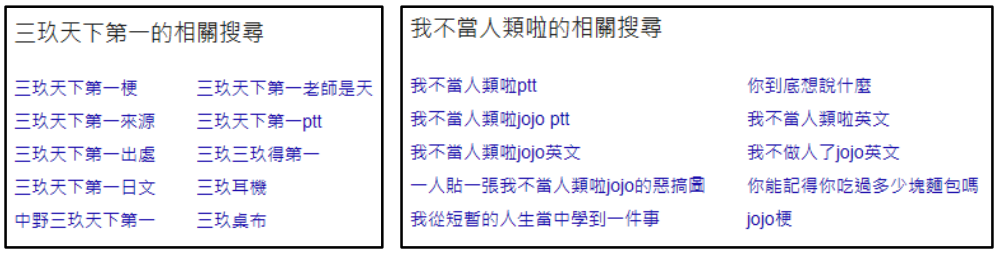
5. Pseudo Relevance Feedback (PRF) [7]

   在第一次檢索完畢後，假設最前面的 n 個文件與 query 具有強烈相關，最後面 m 個則是完全無關，並以它們對 query 的 TF-IDF vector 進行調整後再進行第二次檢索，或者是重複上述步驟多次，再得到最終結果。在本次實作中，我們使用了最前面的 5 個文件來進行調整，文件權重會隨著 rank 的排序逐漸減小，重複檢索 3 次後得到最終的結果，分數明顯比僅僅使用 BM25 進步不少。



6. Query Expansion (QE) [8][9]

有時因為題目本身的 query 及欲檢索的新聞使用的是同義詞組，而非完全一模一樣的詞彙，如果只使用 query 中出現的詞進行搜尋，會使得 recall 不夠高。例如：第六篇新聞中提到前總統陳水扁「已是重病患者...還被關在監獄中」，可知此新聞的主題應是「保外就醫」。然而因為新聞內文並沒有包含「保外就醫」此關鍵字，以此關鍵詞來搜尋時就會漏掉這一篇。Google 搜尋就有使用到類似功能：



Query Expansion 的方法可更加細分為以下幾類。最簡單的方法就是手動加入。以第一道問題為例，我們手動把「外遇、小三、廢除、婚外情、無罪」等關鍵詞加入「通姦在刑法上應該除罪化」中。Word embedding 也可以應用於此：透過 word embedding，可以輕易的尋找與特定字詞最相近的詞彙，我們利用這個特性，先嘗試手動輸入關鍵字，將與其最相近的 n 個詞彙（n=3～6）加入 query，再以 BM25 進行檢索。然而，結果不但沒有進步，反而有些許退步。我們認為，手動的 query expansion 效果較佳，其主要原因是 word embedding 的相似詞仍然不夠精準，可能會找到不相關的詞彙降低 precision。例如：「支持臺灣中小學（含高職、專科）服儀規定（含髮、襪、鞋）給予學生自主」一題中，使用 word embedding 搜尋關鍵詞除了較貼近的關鍵詞如「獎懲、襪禁」外，還會找到「髮式、短褲、鞋襪」等等較為通用的關鍵詞。最後，我們也有嘗試將排名前幾名的文章中，TF-IDF 值最高的字加入 query，但結果仍然是退步的，原因應該是原始 query 以及新 query 之間的權重沒有調整好。

7. TD.csv

在官方提供的 TD.csv 中，有部份 query 與測試資料中的相同，因此我們在以 BM25 及 PRF 檢索完畢後，將這些資料加入最終的結果中：首先以 TD.csv 中的資料中 relevance 不為 0 的資料排在最前面，數量不足 300 的部份則再以檢索結果補足，並以 TD.csv 確認其 relevance 不為 0。

實驗結果與討論

方法	BERT	Conv-Knrm	TF-IDF	BM25
分數	0.00012	0.21761	0.19161	0.22074
方法	BM25+PRF	BM25+TD.csv	BM25+PRF+TD.csv	
分數	0.25824	0.37609	0.41458	

從本次的結果來看，傳統 rule-based 的方法完全勝過 neural network 的方法。一開始我們主要是實驗 BERT 與 Conv-Knrm 等 neural network，然而結果都不盡理想，主要的問題應該是有標注的訓練語料過少，而且要偵測的語意又比一般 information retrieval 要來得複雜許多。後來回歸到最簡單的 rule-based 方法，用 TF-IDF 及 BM25 搭配一些臨時想到的規則測試看看，沒想到結果意外地好，而為了更進一步提升正確率，我們也嘗試了許多文獻中都會提及的 query expansion 和 pseudo relevance feedback。Query expansion 的功效可能是來自於提供了同義詞、新聞相關脈絡的資訊，這些都是比較難由 neural network 在少量的訓練資料中得到的，而這也解釋了為什麼使用 word embedding 的 query expansion 效果不彰：word embedding 只能提供通用、常見的相似詞，卻無法判斷這些詞彙和新聞文章有無關聯。Pseudo relevance feedback 則是利用 feedback 的方式來增加訊噪比，拉開有關、無關文章之間的差距。

結論

在 supervised 的設定之下 neural network 的表現固然亮眼，然而在現實應用的情境之中，往往沒有這麼多標注好的資料讓我們訓練模型，此時傳統的 rule-based 方法反而大放異彩：妥當的中文斷詞，加上簡單的 BM25，搭配 query expansion 和 PRF 等後處理，就可以得到相當不錯的結果。

參考資料

[1] Dai, Zhuyun, et al. "Convolutional neural networks for soft-matching n-grams in ad-hoc search." Proceedings of the eleventh ACM international conference on web search and data mining. ACM, 2018.

[2] Cui, Yin, et al. "Kernel pooling for convolutional neural networks." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

[3] Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).

[4] Nogueira, Rodrigo, and Kyunghyun Cho. "Passage Re-ranking with BERT." arXiv preprint arXiv:1901.04085 (2019).

[5] Ramos, Juan. "Using tf-idf to determine word relevance in document queries." Proceedings of the first instructional conference on machine learning. Vol. 242. 2003.

[6] Robertson, Stephen, and Hugo Zaragoza. "The probabilistic relevance framework: BM25 and beyond." Foundations and Trends® in Information Retrieval 3.4 (2009): 333-389.

[7] Cao, Guihong, et al. "Selecting good expansion terms for pseudo-relevance feedback." Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 2008.

[8] Efthimiadis, Efthimis N. "Query Expansion." Annual review of information science and technology (ARIST) 31 (1996): 121-87.

[9] Kuzi, Saar, Anna Shtok, and Oren Kurland. "Query expansion using word embeddings." Proceedings of the 25th ACM international on conference on information and knowledge management. ACM, 2016.