應用階層可解構式注意力模型於新聞立場辨識任務

A Hierarchical Decomposable Attention Model for News Stance Detection

黃晨郁　Chen-Yu Hunag
國立中央大學資訊工程系
Department of Computer Science & Information Engineering
National Central University
lensixtwo@gmail.com

張嘉惠　Chia-Hui Chang
國立中央大學資訊工程系
Department of Computer Science & Information Engineering
National Central University
chia@csie.ncu.edu.tw

摘要

新聞立場辨識任務的目的為判斷一篇新聞對於某個議題的立場是中立、贊成或反對。此項任務與自然語言推理 (Natural Language Inference, NLI) 任務類似，目標在給定兩個句子，判斷兩者之間是否無關或存在蘊涵、矛盾關係。本論文以新聞立場檢索競賽提供的資料作為參考，但其大部分的新聞文章都屬於支持特定議題的立場，造成資料在不同類別的分佈不平衡。本篇論文提出 Hierarchical Decomposable Attention Model 來解決新聞立場辨識任務，我們以句子為單位分割新聞文章，並基於 Decomposable Attention 的原理找出文章中的每個句子與特定議題的關係。針對資料不平衡的問題，我們建立反義的議題，並手動標記新聞對反義議題的立場，以改善模型效能。實驗結果顯示，我們提出的模型效能優於其他模型。

Abstract

The goal of News Stance Detection task is to detect whether the stance of a news article is neutral, approval or opposition with respect to a given query. The task is similar to Natural Language Inference (NLI) task, which aims to determine if one given statement (a premise)

semantically entails another given statement (a hypothesis). Since most news articles hold neutral stances with respect to the given query, the training data is often unbalanced. In this paper, we proposed a Hierarchical Model based on the Decomposable Attention Model for NLI tasks to compare individual sentences with the given query and jointly predict the stance of the complete article. For the data imbalance problem, we heuristically create opposite queries and label supporting news articles from unrelated ones of the original query to identify unrelated news articles. The experiment result showed that the performance of our architecture is better than other models.

關鍵詞：新聞立場辨識，自然語言推理，篇章分析，注意力機制

Keywords: News Stance Detection, Natural Language Inference, Discourse Analysis, Attention Mechanism.

一、緒論

在新聞立場辨識（News Stance Detection）中，我們必須根據給定議題及新聞文章，去判斷此篇新聞立場為中立或是偏向贊成或是對立。AI CUP 2019 的新聞立場檢索競賽，其目的是開發一搜尋引擎，使用者能根據其輸入的議題，從大量新聞文章中找出與議題相關且立場一致的新聞文章，幫助閱聽人釐清新聞文章所代表的立場，從不同角度去了解各種爭議性的議題。競賽主辦單位提供國內新聞的網頁連結、包含立場的爭議性議題作為查詢題目以及標記好的訓練資料。表一為主辦單位提供的訓練資料範例，每筆資料包含議題、新聞文章以及兩者的相關程度。在評估的部分，主辦單位採用 MAP@300（Mean Average Precision at 300）指標來評估系統效能。MAP@300 的值介於 0 到 1 之間，值愈高表示搜尋結果愈好，其計算方式可參考競賽網站。

目前資訊檢索領域已發展出許多現成的工具，如：Solr 和 Elasticsearch，方便我們根據議題從大量文件中找尋相關的文件。然而，在相關的文件中，我們會發現內容與議題相關，但立場卻相反的新聞文章。如表一的 D1，該內容雖然與「陳前總統保外就醫」的議題相關，但包含了「反對陳前總統保外就醫」的論述，因此該篇文章與「支持陳前總統保外就醫」的議題存在著相反的立場，並且標記為 Irrelevant。

表一、新聞立場辨識資料範例

Query	支持陳前總統保外就醫	
D1	前總統陳水扁申請出席國慶大典昨遭到台中監獄駁回...中監應該考慮撤銷他保外就醫資格，讓他回去服刑。…	Irrelevant
D2	前總統陳水扁在獄中罹患重度憂鬱症、重度阻塞型睡眠呼吸暫止症…扁真的病了，早就應該儘快「保外就醫」…	Relevant

新聞立場辨識與自然語言推理任務的目標相似，藉由輸入兩個訊息後，辨識一個給定的陳述（premise）在語義上是否蘊涵另一陳述（hypothesis）。然而，由於新聞文章屬於由多個句子組成的結構（篇章），因此，在僅知道整篇文章與議題之間的關係，且文章中的每個句子與議題間的關係都是未知的情況下，如何去結合每個句子和議題的蘊涵資訊，以及運用句子間的篇章關係（Discourse Relations）成為本項任務困難之處。另一方面，本論文以新聞立場檢索競賽提供的資料作為訓練模型之參考，但我們發現該資料中，新聞內容與議題的立場不同的資料比例很少，此現象造成訓練模型不易，也是本項任務的困難之處。

根據上述的問題，我們參考了 Decomposable Attention Model[1]，藉由該模型針對輸入的兩個句子編碼生成句子表示 (Sentence Representation)，並結合 Durmus 等人[2]在立場辨識任務中使用了階層式架構的想法，提出 Hierarchical Decomposable Attention Model，階層式分析文章每個句子與議題的關係。針對資料不平衡的問題，我們試圖加入新聞內容與議題立場不相關的資料來平衡資料，藉此提升模型在不相關資料的效能。

二、相關研究

（一） Natural Language Inference (NLI)

在 NLI 相關的研究中，Parikh 等人提出的 Decomposable Attention Model[1]是基於對齊的概念[5]的神經網路架構。該架構包含：Attend、Compare 以及 Aggregate 三個部分，輸入為兩個句子（以a和b表示），輸出為兩者關係預測（以y表示）。首先，在 Attend 的部分對齊（soft-align）a和b中每個詞。接著，在 Compare 的部分會比較原有的句子和上個部分所計算的對齊子句段(aligned subphrase)。最後，在 Aggregate 的部分會結合 Compare 的結果並輸出最後的標記。相較於其他現有的架構，由於 Decomposable Attention Model 所需訓練的參數量少，訓練的時間也較短，但效能卻能超越其他更為複雜的模型。然而，

由於沒有考慮詞的順序，因此在某些情形下會有準確率不高的現象。

（二） Stance Detection

立場辨識（Stance Detection）是根據一段帶有立場的文本，將作者對目標（如：議題或事件）的立場進行分類（如：贊成或反對）的問題。目前與立場辨識相關的任務主要可以分成三種類別[6]，分別是：**Generic Stance Detection**、**Rumour Stance Classification** 以及 **Fake News Stance Detection**。

Generic Stance Detection 的例子為 SemEval 2016 stance dataset[7]，該資料集的每筆資料由一段文字以及目標組成，此任務的目的即是根據以上資訊辨識出作者對於目標的立場是 *favor*、*against* 或是 *neither*。與 Rumour Stance Classification 相關的問題出現在 RumourEval task[8]，該任務必須根據包含謠言（rumour）的來源推文和同一對話線程中的推文，將對話線程中的推文對於謠言的立場分類成 *supporting*、*denying*、*querying* 或是 *commenting*。Fake News Challenge 的立場辨識任務被視為辨識假新聞的首要步驟，其目的是建立一個可評估新聞來源對特定主張所含有的立場的自動化系統。立場分為四種：*agree* 表示文章內容與標題立場一致；*disagree* 表示文章內容與標題立場不一致；*discuss* 表示文章內容討論到標題，但不含有立場；*unrelated* 表示文章內容沒有討論到標題。

Durmus 等人[2]定義了另一種立場辨識的任務：**Claim Stance Detection**。在該任務中，必須根據 Argument Path 上的前後兩個主張，判斷後者的立場是支持或反對前者。實驗結果顯示，透過階層式（hierarchical）而非平坦式（flat）來表示 Argument Path 上的內容可以達到較好的結果。

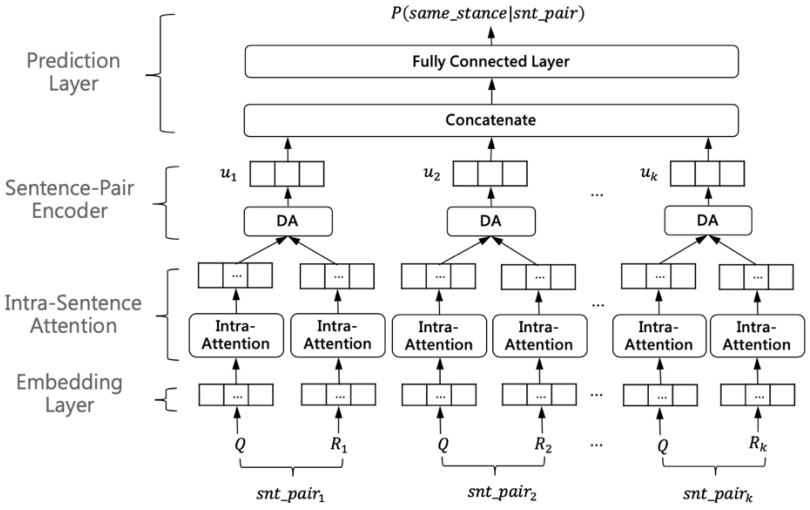
（三） Discourse Parsing

篇章分析是分析篇章中子句之間的層次結構和語義關係，進而了解篇章所要表達的意義。篇章剖析（Discourse Parsing）模型以標記好的篇章剖析語料庫作為訓練資料，如英文的 Rhetorical Structure Theory (RST)[9]和 Penn Discourse Tree-bank Project (PDTB)[10]，前者以樹狀結構（tree structure）來表示篇章中句子之間的關聯，後者以平坦結構（flat structure）表示含有顯式連接詞或隱式連接詞的兩個句子之間的關係；中文的 Chinese Discourse Tree Bank (CDTB)[11]語料庫則與 RST 相似，以樹狀結構來表示篇章關係，並將關係分成：因果（Causality）、轉折（Transition）、解說（Explanation）以及並列（Coordination）。

篇章剖析的過程包含四個步驟：基本篇章單位的切割、剖析樹結構建立、子句關係標記以及中心關係標記。基本篇章單位（Elementary Discourse Unit，EDU），又稱為子句，是篇章剖析樹上的子節點。在建立剖析樹之後，所有子句會以階層式樹狀架構呈現。最後，標記剖析樹中相連子樹之間的關係（Sense Label）（如：並列、因果、轉折或解說）以及中心關係標記（Center Label）（如：核心結構在前、在後或相等）。

三、模型架構

新聞立場辨識的目的為根據給定的文章與議題，辨識兩者的立場是否相同。根據此任務，我們提出 Hierarchical Decomposable Attention Model。首先，基於 Decomposable Attention Model[1]，對輸入的兩個句子進行編碼生成句對表示（Sentence-Pair Representation）。接著，受到 Durmus 等人[2]在 Claim Stance Detection 任務中使用了階層式架構的啟發，我們以階層式的方式處理新聞文章，找出新聞文章中的句子彼此間的關係，進而判斷文章與議題之間的立場是否相同。模型架構如圖一所示，包含了以下部分：**Embedding Layer**、**Intra-Sentence Attention**、**Sentence-Pair Encoder** 以及 **Prediction Layer**。模型的輸入由新聞文章與議題配對而成，輸出為對應的預測標記以表示兩者的立場是否相同。



圖一、Hierarchical Decomposable Attention Model 架構圖

（一） Input Representation

模型的輸入為議題與新聞文章，以(Q, D)表示，文章D包含了n個句子，以$D = \{S_1, \ldots, S_n\}$表示。輸出為兩者的立場是否一致，以y表示。訓練資料以集合$\{Q^{(n)}, D^{(n)}, \mathrm{y}^{(n)}\}_{n=1}^{N}$表示，

其中y ∈ {0, 1}。由於文章平均大約為 53 個句子，加上文章當中可能包含與議題較為無關的部分，因此我們嘗試使用以下設定，從中選取k個句子，以$R = \{S_{f_1}, \dots, S_{f_k}\}$表示，$f_k$為被選取到的句子索引。為了簡化符號，我們令$R_i = S_{f_i}$，並以$R = \{R_1, \dots, R_k\}$表示被選取到的$k$個句子。

1. **Original order:** 根據句子在文章中原有的順序，選取前k個句子。
2. **Cosine similarity:** 根據每個句子和議題的詞嵌入（Word Embedding），計算兩者之間的餘弦相似度，並根據相似度排序，選取前k個與議題相似的句子。
3. **Discourse parsing:** 使用篇章剖析模型 RvNN-CYK2 Seq-EDU + self-attentive Model[12] 處理新聞文章，產生對應的子句、關係標記（Sense Label）和中心關係標記（Center Label）。由於我們認為含有“Causality (因果)”和“Transition (轉折)”關係的句子在文章中含有較為重要的資訊，因此我們從中選取前 k 個被標為“Causality (因果)”和“Transition (轉折)”關係的中心子句。

有了選取好的句子後，我們以兩種設定形成句子配對snt_pair_i，作為模型的輸入：

1. **Parallel:** 議題Q與每個被選取到的句子$R = \{R_1, \dots, R_k\}$配對，形成$\{(Q, R_1), (Q, R_2), \cdots, (Q, R_k)\}$。
2. **Series:** 議題Q僅與R_1配對，被選取到的句子R'則依照順序配對，形成$\{(Q, R_1), (R_1, R_2), \cdots, (R_{k-1}, R_k)\}$

（二） Embedding Layer

針對長度為m議題Q和l的句子R_s，以訓練好的 Word2Vec 模型表示每個詞的向量，分別為$Q = (q_1, \dots, q_m)$和$R_s = (w_1, \dots, w_l),\ \forall s \in [1, \dots, k]$，$q_i, w_j \in \mathbb{R}^{d_w}$，$d_w$為 Word2Vec 模型的維度。

（三） Intra-Sentence Attention

透過計算每個句子中每個詞的權重，找出句子本身重要的詞。如公式(1)(2)所示，c_{ij}為同一個句子的詞向量經過全連階層F_{intra}後彼此內積的結果；c_{ij}透過 softmax 運算，並與原本的詞相量進行加權總合後，可得q_i'和w_i'。在後續的運算中，我們將原有的詞向量與q_i'和w_i'連接，作為新的詞向量，以$q_i := [q_i, q_i']$和$w_i := [w_i, w_i']$表示。其中，$W_1 \in$和$b_1 \in \mathbb{R}$，

皆為模型中可訓練的參數。

$$c_{ij} = F_{intra}(q_i)^T F_{intra}(q_j)$$
$$F_{intra}(q_i)=ReLU(q_i^T W_1 + b_1) \quad (1)$$

$$q_i' = \sum_{j=1}^{m} \frac{exp\ (c_{ij})}{\sum_{k=1}^{m} exp\ (c_{ik})} q_j \quad (2)$$

（四） Sentence-Pair Encoder

我們使用 Decomposable Attention model[1]（以DA表示）作為 Sentence-Pair Encoder，針對snt_pair進行編碼以取得對應的句對表示$u_s \in \mathbb{R}^{d_u}$。最後，我們將u_s向量連接取得向量$u \in \mathbb{R}^{kd_u}$，如以下公式：

$$u_s = DA(snt\_pair_s), \qquad \forall s \in [1, \dots, k] \quad (3)$$

$$u = \left[u_1, u_2, \dots, u_k\right] \quad (4)$$

（五） Prediction Layer

向量u經過 Prediction Layer 計算文章與議題為相同立場的機率P，如下列公式所示，$W_{p_1} \in \mathbb{R}^{kd_u \times hidden}$、$W_{p_2} \in \mathbb{R}^{hidden}$、$b_{p_1} \in \mathbb{R}^{hidden}$以及$b_{p_2} \in \mathbb{R}$皆為模型中可訓練的參數，$\sigma$為激勵函數。

$$P(same\_stance|snt\_pair) = \sigma\left(ReLU(u^T W_{p_1} + b_{p_1})W_{p_2} + b_{p_2}\right) \quad (5)$$

模型在訓練的過程中的損失函數為交叉熵(cross-entropy)，並使用 L2 正規化（權重為 λ），θ為模型中可訓練的參數集合，N為訓練資料的大小。計算方式如下：

$$L(\theta) = \sum_{n=1}^{N} \left[z^{(n)} \cdot log\ P^{(n)} + (1 - z^{(n)}) \cdot log(1 - P^{(n)})\right] + \lambda \|\theta\| \quad (6)$$

四、實驗與分析

在本章節中，將詳細分析本論文所使用的資料，並比較其他架構與本論文提出之模型在新聞立場辨識的效能。最後，我們針對不同的資料輸入形式進行比較，找出合適的方式處理新聞資料。

（一） 資料集

本研究所使用的資料來自 AI CUP 2019 的新聞立場檢索競賽所提供的新聞語料庫（NC）以及訓練語料（TD）。新聞語料庫包含了 80 萬篇新聞編號以及連結，我們根據這些連結爬取了對應的文章。訓練語料包含了 4,662 筆資料，資料範例如表二所示，包含 Query、News_Index、Relevance 三個欄位，Query 為訓練查詢題目，News_Index 為編號，Relevance 為相關程度，相關程度 0、1、2、3 分別代表不相關 (0)、部分相關 (1)、相關 (2)、非常相關 (3)。Query 必定含有立場，若某一文件之內容與 Query 內的議題有關，但立場與 Query 不一致，仍視為不相關 (0)。

表二、訓練語料範例

Query	News_Index	Relevance
贊成流浪動物零撲殺	news_000109	3
核四應該啟用	news_000156	1
遠雄大巨蛋工程應停工或拆除	news_000684	0
拒絕公投通過門檻下修	news_000091	2

表三為在訓練語料所包含文章的相關數據統計。我們使用中研院詞庫小組提供的工具進行中文斷詞，並以“，。、！？”等符號來分隔句子。

表三、訓練語料中的文章的相關數據統計

	Mean
# Sentence / Article	53
# Word / Article	383
# Char / Article	704
# Word / Sentence	7
# Char / Sentence	13

訓練語料中各相關等級所含有的資料個數如表四所示。在實際觀察訓練語料後，我們發現相關程度與一篇新聞中對於議題立場所佔的比例有正向關係，且大部分文章僅含有支持或反對其中一方的觀點。另一方面，該競賽在評估 MAP@300 時認定相關程度在 1 以上即屬於立場相同。因此，本研究參考主辦單位的評估標準，根據立場相同與否來分類新聞文章，而不考慮細部的相關程度。若依照此分類方式，其資料比例如表五的 Original 欄位所示，由於不相關立場的資料比例大約只佔整體資料的 12%，當分成兩類時，即出現了資料不平衡的問題（Data Imbalance Problem）。

表四、訓練語料各類資料的比例

Relevance	#Articles	Pic %
0 (不相關)	546	12%
1 (部分相關)	2071	44%
2 (相關)	1537	33%
3 (非常相關)	508	11%
Total	4662	100%

表五、根據立場相關與否將訓練語料分成兩類後的資料比例

Relevance	Original	Opposite	Original+Opposite
0 (不相關)	546 (12%)	4559 (98%)	4615 (55%)
1 (相關)	4116 (88%)	103 (2%)	4219 (45%)
Total	4662	4662	9324

針對資料不平衡的問題，我們試圖增加標記為「不相關」的資料個數以平衡資料。我們以手動的方式產生與原有議題立場相反的議題（以下稱 **Opposite Query**），將原有訓練語料的 Query（以下稱 **Original Query**）中的一些詞替換成相反意義的詞。例如：「支持」替換成「反對」、「應該」替換成「不應該」…依此類推。所有立場與 Original Query「相關」的文章，對 Opposite Query 的立場變為「不相關」；考慮到對 Original Query「不相關」的文章可能包含立場中立的情形，因此我們利用人工標記的方法，去標記其對 Opposite Query 的立場。如表五的 Original+Opposite 欄位所示，兩種類別的資料比例變得較為平衡。

（二） 實作細節

在資料前處理方面，我們將文章中所有出現的網址以"_url_"符號表示，並移除掉所有中文、英文以及數字以外的符號。在詞嵌入方面，用前處理好的新聞語料庫（NC）作為訓練資料，設定 window size 為 5，並以 Skip-gram 建立維度為 100 的 Word2Vec 模型[12]。

在超參數的部分，輸入的批次量為 32、Epoch 最大為 300、模型在驗證資料的效能若超過 60 個 Epoch 沒有提升，將停止訓練。Hidden Size 為 100、Dropout 為 0.3、學習率為 1e-3、最佳化器使用 Adam、L2 正規化權重 λ 設定為 1e-4。

競賽單位提供的訓練語料中包含了 20 種不同的 Query，為避免相同的 Query 同時出現在訓練與測試資料中，因此我們根據 Query 的不同去分割資料，並重複以下切割方式十次來評估模型：如表六所示，我們隨機選取 Original Query 中 80%的資料以及其 query-article pair 作為訓練資料；測試資料的部分，我們分成兩種測試資料，一種是

Original Query 中訓練資料以外的 20%（以下稱為 **Original Test**），一種則是 Original Test 相對應 Opposite Query（以下稱為 **Opposite Test**）。

表六、訓練與測試資料的分割

Split	# Original queries	# Opposite queries
Train	16 (80%)	16
Test	4 (20%)	4

考慮到資料在兩類別的數量不平衡，我們同時計算了 macro-average 和 micro-average F1-score，並以兩者的平均作為後續實驗的評估方法：

$$Avg\_F = \frac{1}{2}(Marco\_F + Micro\_F) \tag{7}$$

（三） 模型效能比較

本節將針對其他三種架構與本論文提出的模型進行比較，同時比較在加入訓練資料之後對效能的影響。在本節的實驗中，每篇文章被選取的句子個數（以k表示）皆為 5。我們比較的三種架構如下所述。

Decomposable Attention model[1]（以 **DA (Flat)**表示）：原始輸入的句子配對(a, b)改為議題與文章配對(Q, R')，Q為議題，R'由文章R中前k個句子$R' = \{R_1, \dots, R_k\}$串接而成，以 Flat 的形式輸入，議題與文章最大長度皆設定為 100 個詞。

Fine-tuned BERT[14]（以 **BERT (Flat)**表示）：輸入序列為(Q, R')配對之間插入分隔符號[SEP]，並以分類符號[CLS]作為序列開頭，並在[CLS]的輸出位置接上一個線性分類器。議題與文章最大長度皆為 100 個字元。使用預訓練好的 BERT 中文模型，其參數設定 Transformer 層數為 12、Hidden Size 為 768、Multi-Head 數量為 12。

Fine-tuned BERT (hierarchical)[2]（以 **Hi-BERT** 表示）：每對 sentence pair 經過 BERT 後編碼取得 sentence pair representation R_{pairi}，接著，透過雙向 GRU 找出每對R_{pairi}之間的關係，最後，接上線性分類器後輸出預測結果。Bi-GRU 的 Hidden Size 為 100。兩句子合計最大長度設定為 30 個字元（含[CLS]符號與[SEP]符號）。

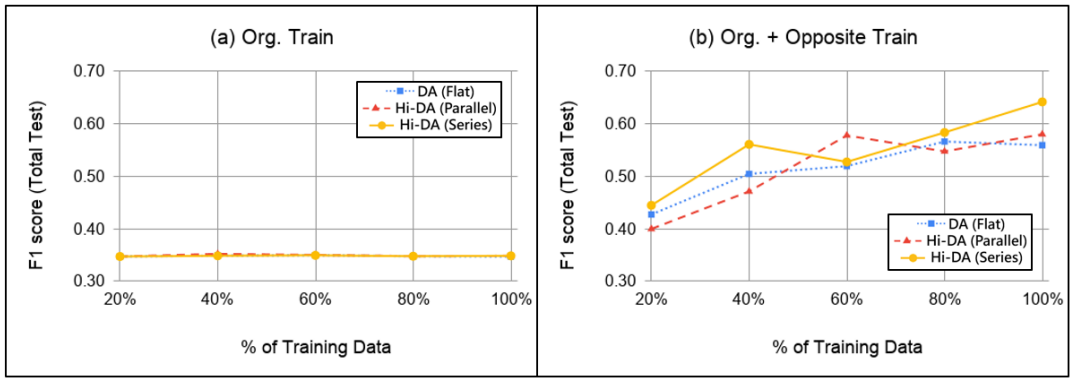
表七為在 Original Train 和加入 Opposite Train 兩種情形下訓練模型後，在 Original Test 和 Opposite Test 兩種資料集測試的結果。首先，比較各模型後，我們發現比起以 BERT 為基礎的模型（BERT (Flat)和 Hi-BERT），以 Decomposable Attention 為基礎的模型

可以達到較好的效能（DA (Flat)和 Hi-DA）。接著，在我們將 DA (Flat)改為階層式的架構之後，模型在兩種測試資料集的效能皆有所提升。最後，可以發現對所有模型而言，若僅以 Original Train 訓練，模型在 Opposite Test 上的效能明顯不佳，我們推論這是由於 Original Train 當中僅有少部分「不相關」的資料，因此模型難以學習如何辨識「不相關」的資料；當我們加入 Opposite Train 訓練後，可有助於模型辨識出「不相關」的資料，因此在 Opposite Test 的結果有明顯提升。

表七、不同模型在加入 Opposite Train 後的效能比較

Model	Avg. F1 score					
	Trained with Org. Dateset			Trained with Org.+ opposite Train		
	Original Test	Opposite Test	Total	Original Test	Opposite Test	Total
BERT (Flat)	0.6556	0.0385	0.3471	0.4319	0.5345	0.4832
Hi-BERT (Parallel)	0.6567	0.0389	0.3478	0.3326	0.5671	0.4498
Hi-BERT (Series)	0.6554	0.0386	0.3470	0.3407	0.5599	0.4503
DA (Flat)	0.6522	0.0498	0.3510	0.6046	0.4973	0.5509
Hi-DA (Parallel)	0.6605	0.0387	0.3496	0.6571	0.5040	0.5806
Hi-DA (Series)	0.6559	0.0380	0.3470	0.6592	0.6245	0.6419

為了進一步驗證加入 Opposite Train 能夠提升效能，我們比較模型在使用 Original Train 以及 Original + Opposite Train 訓練的學習曲線，如圖二所示。在圖二左半部中，所有模型的效能皆沒有隨著資料的增加而上升；在圖二右半部中，各模型的效能整體都比左半部好，同時，相較於其他兩種模型，Hi-DA (Series)的效能大致上隨著資料的增加而上升，並達到比其他兩種架構高的效能。



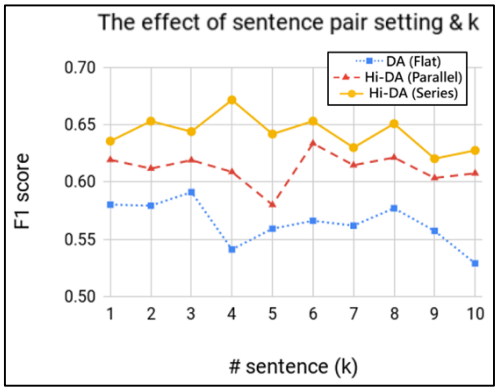
圖二、訓練資料為 Original Train 以及 Original + Opposite Train 的學習曲線

（四） 資料形式的影響

此節將分析我們所提出的從文章中選擇句子的三種方法（以下稱 Sentence Filter）和配

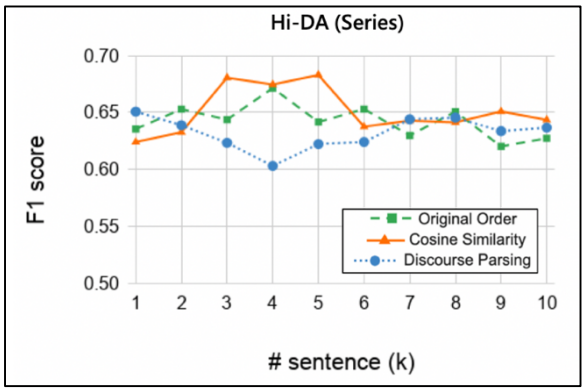
對句子的兩種設定（以下稱 Sentence Pair Setting）對模型效能的影響，並皆以 Original Train + Opposite Train 作為訓練資料。

在 Sentence Pair Setting 比較的實驗中，比較當句子以 Flat 和 Hierarchical 的形式作為輸入後，以 Decomposable Attention 為基礎的模型在測試資料的表現，如圖三所示。可以觀察到當句子以 Series 的方式配對時，結果普遍比其他兩種形式好。顯示出 Series 的方式相較於其他兩種方法，更有助於模型判別文章中的每個句子之間的關聯，並進一步推論文章與議題的立場相同與否。



圖三、不同 Sentence Pair Setting 對模型效能的影響

在 Sentence Filter 比較的實驗中，我們以三種 Sentence Filter：依照句子在文章中原有的順序（Original Order）、議題與句子之間的相似度（Cosine Similarity）以及透過篇章剖析模型處理（Discourse Parsing）等方法從每篇文章中選取k個句子後，比較 Hi-Decomposable Attention（Series）模型在測試資料的表現，如圖四所示。可以發現若以議題與句子之間的相似度（Cosine Similarity）作為選擇句子的依據，當$3 \leq k \leq 5$時，相較於其他兩者方法可以取得較好的結果。因此，我們認為使用相似度（Cosine Similarity）找出和議題較相關的句子，對模型效能的提升是有幫助的。另一方面，我們也發現使用篇章剖析模型處理文章（Discourse Parsing）後無法提升效能的現象。



圖四、Hi-DA (Series)在使用不同 Sentence Filter 後的結果

五、結論與未來工作

本篇論文提出 Hierarchical Decomposable Attention Model，來解決新聞立場辨識任務，同時藉由實驗比較不同的方式來配對和選擇新聞文章中的句子。根據實驗結果，我們提出的模型效能優於其他架構之外，各種模型在加入訓練資料後，效能都有所改善。

本論文提出的模型可與 Solr 或 Elasticsearch 等搜尋平台結合：先透過搜尋平台找出與議題相關的文章後，再以訓練好的立場辨識模型去判別該文章與議題的立場是否一致。但由於我們目前無法取得競賽單位所提供的測試題目對應的標記，因此未來若取得相關標記資料，我們將驗證該模型是否能幫助搜尋平台在立場判別的辨識。另一方面，針對使用篇章剖析模型處理文章後卻無法提升效能的現象，我們提出以下方法改善：第一，透過人工的方式對篇章模型產生的標記進行驗證，再以此資料作為模型的輸入；第二，使用相關的情緒詞典（如：增廣中文意見詞詞典（AUTUSD）[15]）來辨識文章中的句子與特定議題所包含的字詞情感是否一致，再進一步辨識文章與議題的立場。

參考文獻

[1] A. P. Parikh, O. T¨ackstr¨om, D. Das, and J. Uszkoreit, “A decomposable attention model for natural language inference”, in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP2016, Austin, Texas, USA, November 1-4, 2016*, The Association for Computational Linguistics, 2016, pp. 2249–2255.

[2] E. Durmus, F. Ladhak, and C. Cardie, “Determining relative argument specificity and stance for complex argumentative structures”, in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy: Association for Computational Linguistics, Jul.2019, pp. 4630–4641.

[3] S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning, “A large annotated corpus for learning natural language inference”, in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal: Association for Computational Linguistics, Sep. 2015, pp. 632–642.

[4] A. Williams, N. Nangia, and S. R. Bowman, “A broad-coverage challenge corpus for sentence understanding through inference”, in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, Association for Computational Linguistics, 2018, pp. 1112–1122.

[5] D. Bahdanau, K. Cho, and Y. Bengio, “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*.

[6] D. K̈u ̧c̈uk and F. Can, “Stance detection: A survey”, *ACM Comput. Surv.*, vol. 53, no. 1, Feb. 2020, issn: 0360-0300.

[7] S. Mohammad, S. Kiritchenko, P. Sobhani, X. Zhu, and C. Cherry, “Semeval-2016 task 6: Detecting stance in tweets”, in *Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, San Diego, CA, USA, June 16-17, 2016*, The Association for Computer Linguistics, 2016, pp. 31–41.

[8] L. Derczynski, K. Bontcheva, M. Liakata, R. Procter, G. W. S. Hoi, and A. Zubiaga, “Semeval-2017 task 8: Rumoureval: Determining rumour veracity and support for rumours”, in *Proceedings of the 11th International Work-shop on Semantic Evaluation, SemEval@ACL 2017, Vancouver, Canada, August 3-4, 2017*, Association for Computational Linguistics, 2017, pp. 69–76.

[9] W. C. Mann and S. A. Thompson, “Rhetorical structure theory: Toward afunctional

theory of text organization", *Text & Talk*, vol. 8, no. 3, pp. 243–281, 1988.

[10] R. Prasad, B. Webber, and A. Joshi, "Reflections on the Penn discourse Tree Bank, comparable corpora, and complementary annotation", *Computational Linguistics*, vol. 40, no. 4, pp. 921–950, Dec. 2014.

[11] Y. Li, W. Feng, J. Sun, F. Kong, and G. Zhou, "Building chinese discourse corpus with connective-driven dependency tree structure", in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, ACL, 2014, pp. 2105–2114.

[12] Y.-J. Wang and C.-H. Chen, "Using attentive to improve recursive lstm end-to- end chinese discourse parsing", in *The 2019 Conference on Computational Linguistics and Speech Processing ROCLING 2019*, The Association for Computational Linguistics and Chinese Language Processing, 2019, pp. 388–397.

[13] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space", in *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, 2013.

[14] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding", in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186.

[15] S.-M. Wang and L.-W. Ku, "ANTUSD: A large Chinese sentiment dictionary", in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, Portoroˇz, Slovenia: European Language Resources Association (ELRA), May 2016, pp. 2697–2702.