智慧化企業整合

Final Project

透過神經網路處理文本分類問題

108034557 陳宣任

透過建構神經網路模型進行文本分類

清華大學工業工程與工程管理學系 陳宣任

摘要

現今網路蓬勃發展，越來越多人會透過在網路找尋資料或是詢問問題交由網友回覆，而各大論壇也通常會有這樣的功能。然而許多人會利用這種提問的功能，宣傳一些與提問無關的內容或評論，導致垃圾訊息量暴增。但透過人工一一刪除與提問無關的訊息，既沒有效率也無法負荷。因此本研究透過神經網路模型處理文本分類，將問題與其他與提問無關的內容進行分類，而分類完成後，刪除這些無效內容。這個模型可以大幅提高處理效率，同時也能夠提高這些論壇或網頁使用者的滿意度。

關鍵字:文本分類、類神經網路模型、CNNLSTM

1. 背景

1.1. 研究背景

現今網路蓬勃發展，在網路逐漸普及的情況下，網路上的資訊成為了重要的知識來源，也因此越來越多人會透過在網路找尋資料或是發問問題交由網友回覆，來解決各種不同的問題。像是 Yahoo 奇摩的知識家、Facebook 等等，除此之外各大論壇也通常會有這樣的功能。

1.2. 研究動機

隨著這些社群平台的興盛，也有許多人會利用這種發問功能作為自身的宣傳手法或是宣揚自身的見解。造成了有些真正的問題因為被這些內容所掩蓋而無法得到回覆。平台管理者也無法只依靠人力判斷、刪除就能夠清除。除此之外也要避免正確的問答被錯誤刪除，這兩個問題都會導致平台使用者的負面觀感。，

1.3. 研究目的

因此為了解決人工判斷及刪除效率不佳的問題，本研究首先透過 5W1H 分析法了解問題及確認問題解決方式後，在使用類神經網路模型針對文本進行分類，並根據模型分類結果進行訓練與改善，最終得到分類效果最好的模型，提供給平台的管理者。

1.4. 5W1H 分析法

本研究透過 5W1H 分析法，幫助我們了解問題並思考解決問題的方法，對選定的項目、工序或操作，都要從對象（何事 WHAT）、時間（何時 WHEN）、人員（何人 WHO）、地點（何地 WHERE）、原因（何因 WHY）、方法（何法 HOW）等六個方面提出問題進行思考，且對問題進行綜合分析研究，從而得到更具建設性設的決策。利用此方法，我們主要想針對提問垃圾訊息過多的問題來進行改善。

(1)What?: 要解決甚麼問題? 解決平台提問垃圾訊息過多的問題
(2)When?: 在甚麼時候要處理垃圾訊息? 使用者發布訊息至平台後
(3)Who?: 由誰來改善這些問題? 平台的管理者、工程師
(4)Where?: 在哪種地方可以進行問題改善?平台網站
(5)Why?: 為什麼要進行改善? 提高使用者滿意度，讓使用者持續使用此平台。
(6)How?: 如何解決此問題? 建構神經網路模型，區分使用者的提問是否為垃圾訊息。

2. 文獻回顧

2.1. 卷積神經網路(Convolutional Neural Network, CNN)

卷積神經網路是一種前饋式的神經網路，其發展最早由 Hubel 和 Wiesel 在研究貓的視覺皮層時發現了特殊的神經網路架構能夠降低神經網路的複雜性，後續進而提出了卷積神經網路。直到 1989 年 Yann LeCun 所提出了 LeNet-5，其為第一個卷積神經網路框架，如圖 2-1 所示。此框架包括了卷積層、池化層以及全連接層，與現今的卷經神經網路結構相似。而卷積網路架構特別在辨認圖像及影像時，透過特徵擷取，可以得到非常好的成效。

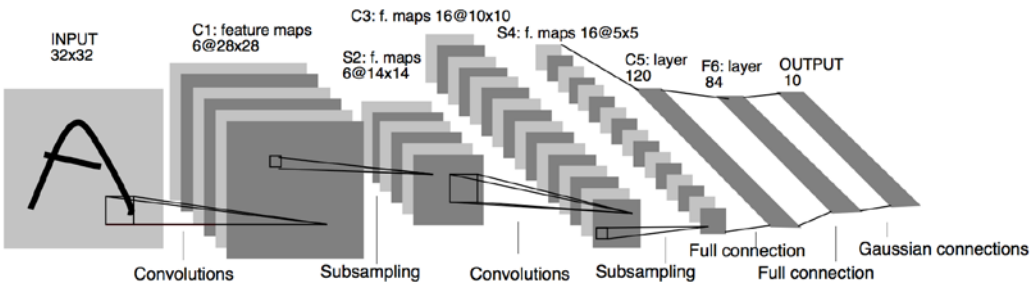

圖 2-1 LeNet-5 架構(Yann,1998)

2.2. 遞歸神經網路(Recurrent Neural Network, RNN)

遞歸神經網路不同於一般的神經網路，其能夠處理具有序列關係的數據，Elman 在 1990 年提出了簡易的 RNN 架構，如圖 2-2 所示。因此常被用於自然語言處理(Natural Language Processing, NLP)。但 RNN 模型會隨著序列的增長，會有著無法學習到序列較前的問題，這在 Hochreiter (1991) and Bengio, et al. (1994) 兩篇文獻當中有提到這樣的情況。

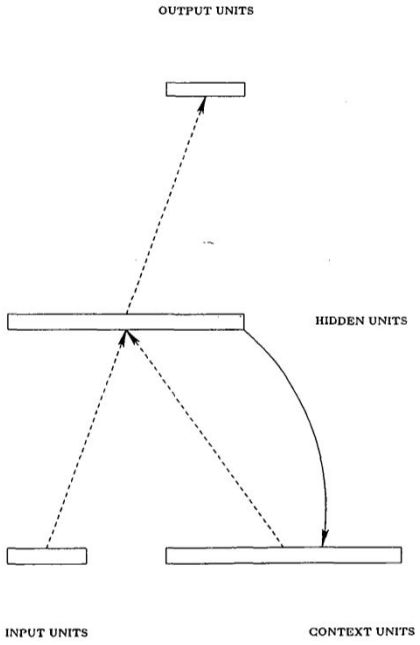

圖 2-2 RNN 基本架構(Eleman,1990)

2.3. 長短期記憶網路(Long Short-Term Memory, LSTM)

長短期記憶網路是一種特殊的 RNN 模型，其最早由 Hochreiter, Sepp, and Jürgen Schmidhuber 在 1997 年時提出，解決了 RNN 在處理長序列數據時，會有梯度消失或梯度爆炸的問題。其在神經單元中加入了”閥(gate)”的概念。用來控制有多少的數據需要被遺忘或是更新。分別為輸入閥(input gate)、輸出閥(output gate)以及遺忘閥(forget gate)，如圖 2-2 所示，透過這樣的方式，可以避免掉這樣的問題。

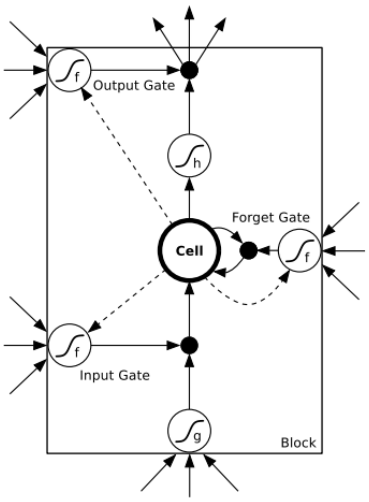

圖 2-2 LSTM 基本架構

3. 研究架構或方法

3.1. 研究架構

本研究之研究架構，大致可分為四個步驟:(1)資料輸入、(2)資料前處理、(3)模型建構、(4)輸出結果、(5)模型訓練，最後得出分類成效最佳的模型。

3.1.1. 資料輸入

本研究透過 kaggle 線上開放式資料庫所提供之 Quora 美國問答網站的使用者提問資料共約 130 萬筆，其中約 121 萬筆為正常提問，9 萬筆為垃圾訊息。

3.1.2. 資料前處理

本研究所做之資料前處理包含下列步驟:

(1) 將正確提問做為第 0 類，垃圾訊息為第 1 類。
(2) 首先將 137 萬筆資料先分出 10%做為測試集，共約 13 萬筆資料。
(3) 而剩餘的資料由於資料不平衡，會影響到訓練的準確性，因此本研究首先透過過採樣及欠採樣的方式，從剩餘的資料當中正常提問與垃圾訊息各取出 10 萬則提問，共 20 萬筆資料。
(4) 將這 20 萬筆資料，拆分為 90%做為訓練集，共約 18 萬筆。10%做為驗證集，共約 2 萬筆。
(5) 補齊 NA 值後，將提問的內容中出現頻率最高的 90000 個字詞轉為向量，並過濾掉特殊字詞。
(6) 將訓練集與測試集的提問設定句子長度為 70 個字詞，也就是超過 70 個字詞

則截斷，低於 70 個字詞補齊。

3.1.3. 模型建構

本研究為了找到最適合文本分類的模型，因此共建構了三種不同的神經網路模型，包括卷積神經網路、長短期記憶網路以及卷積神經網路和長短期記憶神經網路的結合。透過建構三種不同的模型進行訓練，找到最佳模型後再進行參數的修正與改進。首先，三種模型投入層皆為嵌入層，透過詞嵌入的方式將 90000 維的向量轉為 1200 維的向量，而後續各個模型皆有不同之處，本研究分別列出各個架構如圖 3-1 到 3-3 所示。

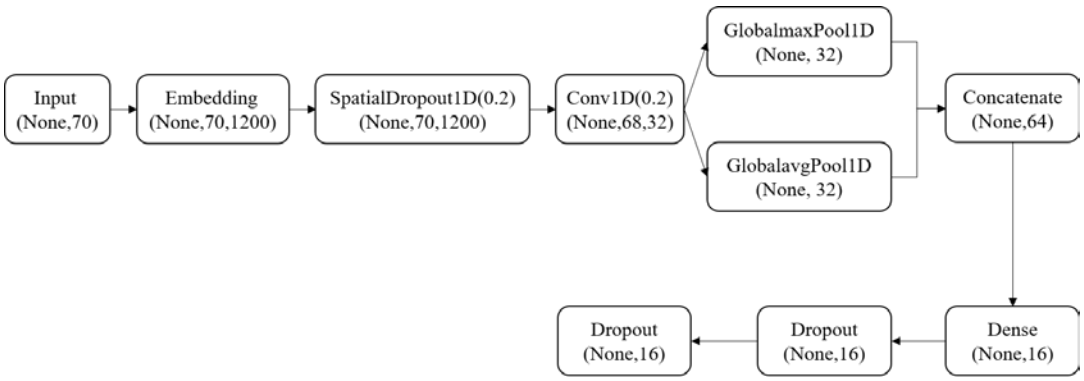

圖 3-1 卷積神經網路架構

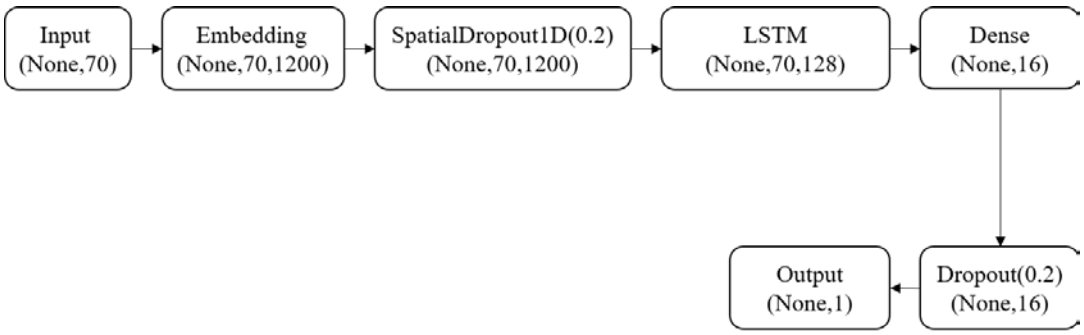

圖 3-2 長短期記憶網路架構

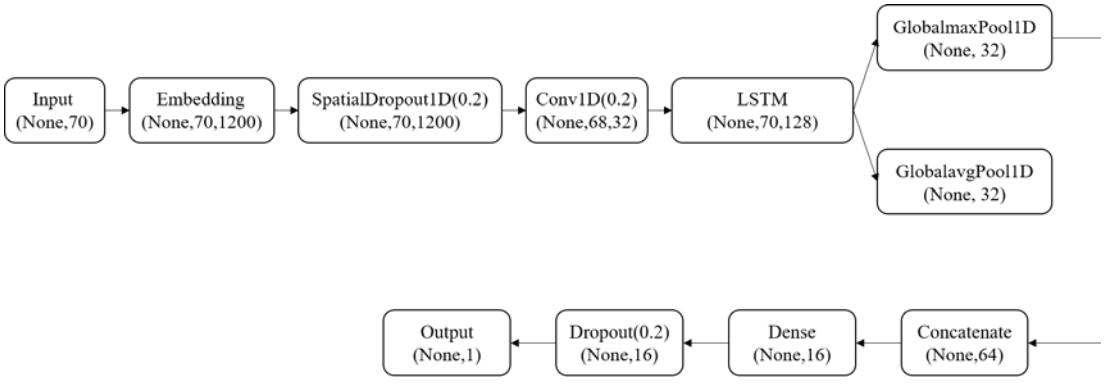

圖 3-3 卷積神經網路與長短期記憶網路架構

3.1.4. 訓練方法設定

模型建構完成後，本研究將三個不同模型分別進行訓練，比較各模型的分類結果。訓練過程激活函數使用 relu 及 sigmoid，損失函數使用二元交叉熵，優化器使用 adam，批量大小為 5000，跑 10 次 Epoch，做為模型的超參數。除此之外，透過提前終止的方式，透過監控驗證集的損失，避免過度擬合的狀況發生。

3.1.5. 輸出結果

在測試集資料不平衡的情況下，單看準確率並無法完全代表模型的好壞程度，必須同時考慮精確度以及召回率，可以做為評估模型好壞的依據。三個模型的分類結果如下所示:

(1) 卷積神經網路模型(CNN)

測試集準確率(Accuracy)0.9121、精確度(precision)0.6894、召回率(recall)0.8651

(2) 長短期記憶網路(LSTM)

測試集準確率(Accuracy)0.8965、精確度(precision)0.6684、召回率(recall)0.8648

(3)卷積神經網路與長短期記憶網路混合(CNN+LSTM)

測試集準確率(Accuracy)0.9140、精確度(precision)0.69、召回率(recall)0.8686

根據三個模型的結果，可以看出表現最差的為 LSTM 模型，而 CNN 與 CNN+LSTM 模型表現相差不大，但 CNN+LSTM 的各項指標仍較高。

3.2. 模型改善

本研究從輸出的結果得到最佳的架構為卷積神經網路與長短期記憶網路混合架構，因此下一步根據此模型去進行改善，可分為下列方面幾種方法。

3.2.1. 增加特徵數

本研究原先是從提問當中擷取頻率最高的 90000 個字詞，但仍會有一些可能是辨別是否為垃圾訊息的字詞沒有被截取到，導致分類不夠準確，因此將最大特徵數提高為 100000 個字詞，測試集準確率 0.9132，精確度 0.701，召回率 0.8725，雖然準確率有一些下降，但精確度及召回率有所上升，且 100000 個字詞基本上應不會比 90000 個字詞差，因此選擇增加特徵數作後續的改善。

3.2.2. 雙向 LSTM

本研究最初使用的皆為單向的 LSTM，這種處理序列的方式只能依據之前時刻的訊息來預測下一時刻的輸出，但在處理文本時，輸出不一定只有與之前時刻的訊息有關，還有可能與未來的狀態有關係，也就是透過上下文來判斷原比只看上文判斷準確率還來的更準確，因此本研究將單向的 LSTM 改為雙向，測試集準確率 0.9121、精確度(precision)0.693、召回率(recall)0.8874，透過雙向的 LSTM，準確率降低，但召回率提高了，也就是實際上是正確提問的內容，被誤分為垃圾訊息的比例降低了。後續本研究單向與雙向皆進行測試，雙向的 LSTM 表現皆較佳，因此雙向 LSTM 仍有其幫助。

3.2.3. 增加卷積層數

在做文本分類時，CNN 扮演了擷取單一句子重要特徵的角色，而每次要擷取劇中的幾個字詞會影響到分類的準確程度，因此應加入不同大小的 filter，做為卷積層，確保能學習到大部分的重要特徵。本研究原先模型只有一層卷積核大小為 3 的卷積層，在經過測試後，分別增加了卷積核大小為 1,2 和 5 的卷積層，因此共有 4 層卷積層，測試集準確率為 0.9286、精確度 0.704，召回率為 0.8912。

4. 研究結果

4.1. 最終模型架構

本研究最終的模型架構如圖 4-1 所示，其中包括一層投入層、一層嵌入層、分別做卷積核大小為 1.2.3.5 的卷積層及池化層，最終加入雙向的 LSTM、以及全連接層，連結到最終輸出。

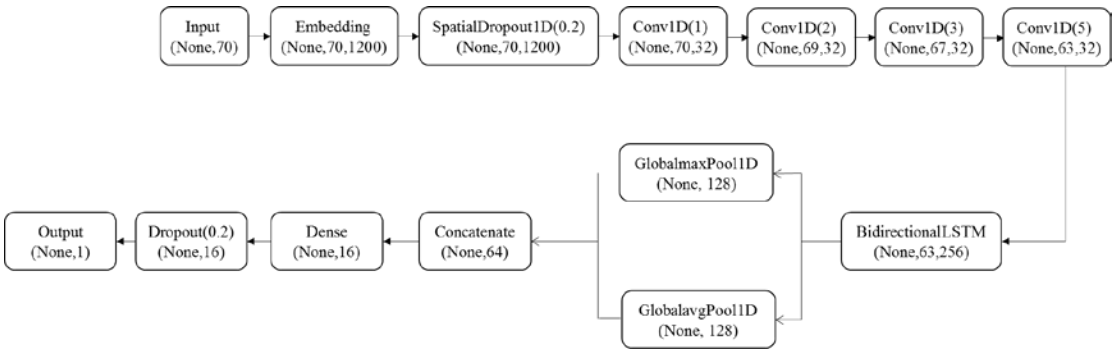

圖 4-1 最終架構

4.2. 最終結果

透過改善過後的模型，可以得到測試集準確率為 0.9286、精確度 0.704，召回率為 0.8912。比起原先的測試集準確率(Accuracy)0.9140、精確度(precision)0.69、召回率(recall)0.8686，三個指標皆有提升，且在本研究所上網搜尋之 20 個樣本作為第二次測試，準確率也有達到 0.75。

5. 結論

5.1. 成果貢獻

本研究透過比較不同模型以及改善方式，得到了最終準確率以及精確度、召回率最高的模型，做為文本分類的方法。未來若有文本分類的問題或是 CNNLSTM 模型的建構方式以及改善方法，可以做為參考的依據。

5.2. 研究限制

本研究雖然有透過 resample 的方式，將訓練集去做平衡，但仍無法達到很好的效果，在測試集中，預測是垃圾訊息，實際上也是垃圾訊息的比例較低，也就是說，有許多模型預測屬於正確提問的文本，事實上為垃圾訊息，因此分類的精確度(Precision)不高，約只有 7 成左右。

5.3. 未來改善

為了可以考慮增加垃圾訊息的樣本數，讓神經網路能夠準確地去分類，才能夠使精確度(Precision)提升。

5.4. 應用

未來可以考慮將此模型運用在其他文本分類上，例如:根據每本書的緒論來將書的類型做分類、檢查書寫內容是否符合規範或是將信箱的信件進行分類等等，應用非常的廣泛。除此之外，CNNLSTM 模型還可以使用在生成一序列圖片或影像的文字描述、影片分類等等，都有非常好的效果。

6. 參考文獻

1. Elman, J. L. (1990). Finding structure in time. Cognitive science, 14(2), 179-211.
2. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.
3. Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. IEEE transactions on neural networks, 5(2), 157-166.