政府部門**QA**問答機器人

李信鋌
110522130
資訊工程學系碩士班

連育陞
110522063
資訊工程學系碩士班

林右展
110522112
資訊工程學系碩士班

1 Literature Survey

在Literature Survey裏面，我們將會講述目前研究者是怎創建和評估他們的聊天機器人，並給出一些例子。

1.1 FLOSS FAQ chatbot project reuse: how to allow nonexperts to develop a chatbot

有研究者指出目前比較常見的聊天機器人為FAQ Chatbot，它們必須為特定的服務，平臺和系統常見問題提供答案（也是我們這次Final Project所要創建的Chatbot類型）。而這種Chatbot需要由多專家組成的工程團隊才能夠實現，例如：Dialog Designer(對話的設計師：負責分析人類的對話，並嘗試用圖表等方式來描述和定義出人類對話的方式）；Experts from the domain（該領域/平臺的專家：負責對該平臺或領域的一些常見問題提供解答，還有負責想額外的問題）；software engineer（軟體工程師：負責聊天機器人的程式部分）。所以作者們了減少聊天機器人的人力和資源消耗，提出了其實不同聊天機器人所需要的資源其實都差不多，在創建新的聊天機器人的時候可以重用某一部分的資源。在這篇論文中，他們在創建聊天機器人的時候重用了NLP模型訓練的資料集，以及把自動化回答的部分都利用程式進行取代。(de Lacerda and Aguiar, 2019)

1.2 Different measurement metrics to evaluate a chatbot system

有研究者提出對聊天機器人進行評分的方法，例如：在這篇文章中，作者提出了一般人在評估聊天機器人是否有效的時候，都會采用Loebner Prize evaluation methodology（羅布納獎）：它的評估方式是採用標準圖靈測試，評審同時與使用電腦的人類及聊天機器人進行文本對話，評審需要根據它們的回應來判斷到底哪一個是人類與聊天機器人，但是研究者覺得這種方式無法評估聊天機器人是否可以就著人類的問題回答出正確的答案，於是他們提出三種另類的方式來評估聊天機器人，分別是：Dialogue Efficiency Metric(對話效率矩陣）：聊天機器人有沒有回答用戶提出的問題；Dialogue quality metric（對話質量矩陣）：聊天機器人有沒有回答出正確的答案，設計者可以根據這個指標來評估機器人答案的準確度；Users satisfaction(使用者滿意度）：使用者是否滿意機器人的回答。設計者可以根據這三種方式來評估他們的聊天機器人到底需不需要額外的改進，如果要改進的話，他們也可以參考這三個矩陣的數據，決定要從哪個方面進行改進。(Shawar and Atwell, 2007)

1.3 Supervised Term Weight Training for Improving Question-Knowledge Matching in Chatbots

這篇為了改善Chatbot system-Ali-me，而設計了多個訓練的加權計算方式，用來改善問題及回答的配對工作，考慮到需要高度的QPS (Query Per Second)，所以他們選擇以較傳統的以回歸模型來做配對，而非深度模型。

為了節省時間，他們並沒有計算問題與每一個回答之間的相似程度，他們利用Lucene toolkit為問題及答案建立索引，並利用TF-IDF 演算法(term frequency-inverse document frequency) 呼叫最好的K 值作為候選答案。

再來才是句子之間的相似度計算，他們做了兩種方式，都是基於訓練後的加權結果來做計算，來獲得問題與候選達案的相關程度。第一，利用加權分數的平均，將句子轉換成向量後，計算雙方的Cosine similarity。第二種，利用Word Similarity Maximization (WSM)，作為多要的Word Mover's Distance (WMD) 並將標準化限制在向量[0,1]，藉以取代未標準化的值。

他們除了利用上述的兩個相似度計算外，也使用了多個boolean值當作特徵值，用做進行估算的依據。而回鍋模型則是使用XGBoost裡的gradient boosted decision tree (GBDT) 來做開發，他們認為有比較好的結果。(Song et al., 2020)

2 Dataset

資料來源有三種方式，分別為

1. 臺北市政府在2017年舉辦的政府部門問答機器人比賽所發放的資料集
2. 從政府資料開放平臺中所找的政府部門常見QA
3. 使用爬蟲，去爬取其他政府部門的常見QA

這三種資料來源會在下文詳細介紹。

2.1 臺北市政府**Chatbot**比賽資料集

第一種資料來源是臺北市政府在2017年舉辦的政府部門問答機器人比賽所發放的資料集，原始的問答是放在台北市政府的網站上:常見問答。這邊一共有從2007 到2017所累積的8512筆常見問答，有229個政府部門。透過人工整理後，可使用的資料筆數是7986筆，包含149個政府部門。圖1為還沒經過處理的資料集格式和容。(臺北市政府常見問答, 2017)

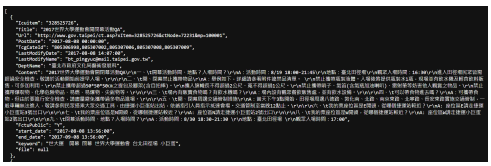

圖 1: 原來的資料集格式

在經過處理後，我們會用來進行Fine-tune的資料格式是：第一個Column為政府部門，第二個Column為和該政府部門有關的問題。圖2為例子。

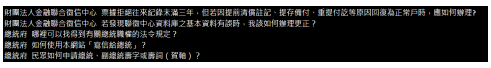

圖 2: 整理過後的資料集格式

2.2 政府資料開放平臺中找到的常見**QA**

第二種資料來源是政府資料開放平臺，它是台灣政府了提升施政透明度，並提升民眾參與公共政策議題的願意程度所設定的一個網站。每個台灣的政府部門都會開放和整理它們部門的資料，並上傳到這個網站上，一般的社會大可以在這個網站拿到相關政府部門的一些資料。我們在這邊主要是搜尋中央政府部門的常見問答資料，最後是得到28個政府部門的常見問答資料，最後經過整理後，資料筆數是5594。

政府部門	資料筆數
內政部人事處	9
內政部中央警察大學	195
內政部戶政司	200
內政部民政司	154
內政部合作及人民團體司籌備處	182
內政部地政司	356
內政部地政機關	278
內政部役政署	275
內政部法規委員會	8
內政部空勤總隊	103
內政部建築研究所	110
內政部政風處	7
內政部消防署	109
內政部秘書室	32
內政部移民署	453
內政部統計處	11
內政部部長室	1
內政部訴願審議委員會	11
內政部會計處	59
內政部資訊中心	167
內政部營建署	475
內政部總務司	111
內政部警政署	482
科技部	239
總統府	37
勞動部勞工保險局	803
經濟部智慧財產局	670
財團法人金融聯合徵信中心	57

表 1: 政府資料開放平臺資料統計

相關的資料放在表1。部分資料集的容會放在圖3和圖4。(政府資料開放平臺, 2013)

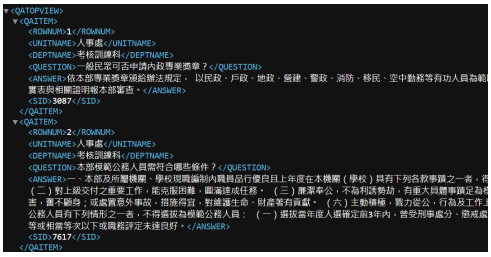

圖 3: 政部資料集

圖 4: 總統府資料集

2.3 使用爬蟲所得的常見QA

第三種資料來源是直接去政府部門的網站，並把它們的常見問答那部分的容和問題都利用爬蟲爬下來，我們使用這個方法來爬取了8個政府部門的常見問答，資料總筆數為1241，相關資料放在放在表2。

政府部門	資料筆數
勞動部職業安全衛生署	193
國家通訊傳播委員會	281
勞動部勞動力發展署	86
勞動部勞動基金運用局	21
衛生福利部社會及家庭署	54
衛生福利部疾病管制署	317
外交部領事事務局	147
教育部	142

表 2: 使用爬蟲所得的資料統計

2.4 資料集版本

而我們的資料集有4個版本。

第一個版本的資料集為經過整理後的臺北市政府Chatbot比賽資料集，裏面有7986筆資料，149個政府部門。我們使用這一份資料集作Chatbot的Baseline。(臺北市政府比賽資料集)

第二個版本的出現原因是希望可以增加更多的政府部門，資料集為經過整理後的臺北市政府Chatbot比賽資料集在加上從政府資料開放平臺中找到的常見QA而成，裏面有12776筆資料，政府部門的數量為180個。（自己整理的資料集v2）

第三個版本的出現是因使用第二個版本來進行Fine-tune的準確度太低了，然後發現資料集有些相關問題和政府部門太相近了，用人工都不太能準確判斷，因此這份資料集只有使用在政府資料開放平臺和爬蟲所得的資料，有35個政府部門（本來是由36個部門，只是我們把內政部部長室的資料去掉，因這個部門只有1筆資料，難以進行訓練和測試），資料筆數為6312。（自己整理的資料集v3）

第四個版本的則是把第三個版本的資料集在加上一些臺北市政府特定的政府部門而成。總共有60個政府部門，資料筆數為7907。（最終整理的資料集）

3 Experiment Design

第一步：收集一些和其他政府部門相關的相關問答。

第二步：把在第一步收集的資料進行預處理，處理的格式為政府部門-關於該政府部門的相關問題，格式的例子可以在圖2看到。

第三步：把處理好的資料變成和BERT和ALBERT相容的格式，把資料集切成訓練集，驗證集，測試集（比例是5：3：2），並使用BERT和ALBERT進行Fine-tune，並評估這些模型的準確度。

Learning Rate為5e-6，並且會自動調整，Training Epoch為30。

BERT所使用的預訓練模型為Transformers套件的bert-base-chinese。(base chinese, 2018)。

ALBERT所使用的預訓練模型為albert-chinese-base。(voidful/albert-chinese base, 2020)。

3.1 BERT

BERT的全名是Bidirectional Encoder Representations from Transformers。它是一個預訓練的語言表徵模型。它强調了不再像以往一樣採用傳統的單向語言模型或者把兩個單向語言模型進行淺層拼接的方法進行預訓練，而是採用新的masked language model（MLM），以致能生成深度的雙向語言表徵，並且在預訓練後，只需要添加一個額外的輸出層進行fine-tune，就可以在各種各樣的下游任務中取得state-of-the-art的表現。在這過程中並不需要對BERT進行任務特定的結構修改。(Devlin et al., 2018)

3.2 ALBERT

ALBERT則是BERT比較小的版本，它會出現的原因是：雖然BERT在很多NLP的任務中取

得成功，但是它的模型架構太大了，導致了它的訓練和預測的速度不夠快，難以應用在講求速度的實際應用上，例如：聊天機器人。所以有研究者研究出ALBERT，它利用了參數共享、矩陣分解等技術大大減少模型參數，利用改進的SOP Loss取代NSP Loss提升了下游任務的表現。因此ALBERT的訓練時間也的確比BERT來得少，並且也可以擴展到比BERT更大的模型，使得ALBERT在各種下游任務的表現不會比BERT差太多，甚至比BERT來得更好。(Lan et al., 2019)

第四步：把Fine-tune好的模型包裝成一個Line Chatbot，讓使用者可以在Line裏面輸入詢問Query，Chatbot根據Query輸出答案。

使用Line的是因它是台灣目前比較常用的通訊軟體是，把聊天機器人架設在Line的話，可以最大化去吸引台灣人來使用這個聊天機器人，從而讓我們在更加容易收集到Dialogue quality metric（對話質量矩陣）所需要的資料，並計算出分數，確認聊天機器人在哪些方面需要進行改進。

4 Result

表3為我們Fine-tune之後的準確度。

資料集	模型	Accuracy
臺北市政府資料集	BERT	75.5
自己整理的資料集v2	BERT	6.3
自己整理的資料集v3	BERT	88.3
自己整理的資料集v3	ALBERT	74.5
最終整理的資料集	BERT	83.3
最終整理的資料集	ALBERT	68.8

表 3: Fine-tune的結果

圖5，圖6為Chatbot的運作情況。

5 Future Work

以下為一些我們在未來可以進行的事項:

1. 增加更多政府部門的資料。讓模型可以辨識更多的政府部門
2. 使用其他的預訓練模型來進行Fine-tune，不一定只使用BERT或者是BERT的變形模型
3. 把Chatbot架設在其他通訊軟體（Telegram，Signal，WhatsApp等等），讓使用其他通訊軟體的使用者也可以使用這個Chatbot

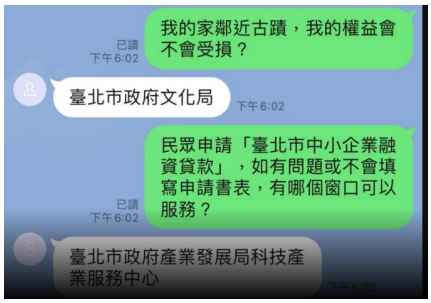

圖 5: Chatbot運作情況

圖 6: Chatbot運作情況

4. 提供一個表單，可以讓使用者輸入他們對於Chatbot的意見，以便評估Dialogue quality metric（對話質量矩陣）的分數。

References

Hugging/bert base chinese. 2018. Huggingface/bert-base-chinese.

Arthur RT de Lacerda and Carla SR Aguiar. 2019. Floss faq chatbot project reuse: how to allow nonexperts to develop a chatbot. In *Proceedings of the 15th International Symposium on Open Collaboration*, pages 1–8.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Bayan Abu Shawar and Eric Atwell. 2007. Different measurement metrics to evaluate a chatbot system. In *Proceedings of the workshop on bridging the gap: Academic and industrial research in dialog technologies*, pages 89–96.

Shuangyong Song, Chao Wang, Haiqing Chen, and Huan Chen. 2020. *Supervised Term Weight Training for Improving Question-Knowledge Matching in Chatbots*, page 13–14. Association for Computing Machinery, New York, NY, USA.

voidful/albert-chinese base. 2020. voidful/albert-chinese-base.

政府資料開放平臺. 2013. 政府資料開放平臺.

臺北市政府常見問答. 2017. 臺北市政府常見問答faq open data.