藉由飯店評論預測旅客旅遊類型之系統

指導教授: 陳建錦教授

Department of Information Management, National Taiwan University, Taiwan

余知諺
Department of Information Management, National Taiwan University, Taiwan

陳映樵
Department of Information Management, National Taiwan University, Taiwan

黃思凱
Department of Information Management, National Taiwan University, Taiwan

ABSTRACT

網路普及帶動了人與人在網路上文字的互動，而這些大量文字、文本中隱藏了一些關鍵的資訊是單靠人為閱讀無法輕鬆得知的。本次專案在於使用 ML 的方式找出評論當中的特徵，希望能藉由評論特徵預測出該則評論是在描述什麼樣的旅遊類型，並進一步去探討不同旅遊類型所在意的層面是否有差異，並進行分析，而此研究中 Feature engineering 的方法主要透過 Bag of Words 的 TFIDF 和 Word Embedding 兩大類的方式來做 Feature engineering，並使用 NB、Logistic Regression、NN 和 LSTM 作為分類模型，透過計算和比較準確率以及 F1 scores 來判斷系統模型的好壞。本次實驗所採用的資料集為 kaggle 上 booking.com 的 515K hotel reviews in Europe，並使用商務旅遊和休閒旅遊作為分類的項目，而顧客所在意的層面則分為設施、交通、服務、食物。

INTRODUCTION

線上整合訂房平台在近年趨於流行，平台上存在著非常多的使用者評論，也意即有著相當可觀的數據量可供行銷方面的資料科學做研究，而我們最主要的就是藉由數據分析技術，從評論內容中萃取出使用者數據用來分析顧客行為，藉此創造商業價值。

而訂房之顧客可以旅遊目的分為商業旅遊與休閒旅遊，在多數飯店或旅遊評論平台上之顧客其實並不會提供自己的旅遊目的，因此會使得平台方與飯店無法針對旅遊類型方面進行分析，而我們的實驗就是在找到適合的評論資料後透過訓練的方式找出最準確的模型來辨別旅遊類型，藉此提供給其他平台甚至是飯店方能更理解不同旅遊類型旅客的需求，另外也是希望可以能用個人較難直觀判斷的類別(旅遊類型)來進行實驗，而非單純的判斷評論的正負面程度或是分數的預測。

本次專案的目標在於找出表現最好的模型能精準分類出 business trip(商業旅遊)和 leisure trip(休閒旅遊)的評論。目前常見的文字特徵工程的方法有 Bag of Words, Bag of N-Grams, TF-IDF, Word embedding，透過這些方法來進行 Feature engineering，而常見的分類模型則有 NB, Logistic Regression, NN, LSTM，其中不管是機器學習或是神經網路我們都有採用，NB 與 Logistic Regression 是屬於機器學習的模型，而 NN 與 LSTM 則是屬於神經網路的模型。

而另外我們的專案在進一步分析不同旅遊類型所在意的層面是否有差異，這樣的分析結果可同時為訂房服務平台與飯店經營者雙方都帶來商業價值:

(1) 針對飯店經營者來說，了解不同類型旅客之需求、可以藉此應用在淡旺季的經營策略。
(2) 針對訂房服務平台來說，了解哪些飯店較受哪種旅遊類型的人青睞、能夠更加優化平台上的推薦系統。

RELATED WORKS

此段落簡述與我們專案實驗相關且在文字分類中有做特徵擷取方式和處理不平衡文本資料集的實驗

在文本的特徵擷取方面，最常見的固定長度功能之一就是 Bag of Words。然而，BOW 特徵卻有個主要缺點：失去了單詞的順序，同時也忽略了單詞的語義。”在這篇論文中[0]他們有別於以往多數實驗是探討 word2vec 的特徵，提出了段落向量(Paragraph Vector)的新特性，一種無監督算法，可以從可變長度的文本片段（例如句子，段落和文檔）中學習固定長度的特徵表示。其目的是克服 Bag of words 模型的弱點。結果表明，Paragraph Vector 運行良好，誤差率相對提高了 32％。段落向量方法明顯優於單詞和雙字母組的事實表明它對於捕獲輸入文本的語義很有用。除此之外， 從大量未註釋的文本中學習主觀名詞列表。然後在一小組註釋數據上訓練主觀性分類器，使用主觀名詞作為特徵以及一些其他先前識別的主觀特徵。而實驗結果表明，主觀性分類器表現良好（77％的 recall，81％的 precision）[3]

另外也因為要從龐大的資料著手，feature 擷取方式也不能很確定對於目標 label 的影響程度為何，同時也因為維度龐大多次造成記憶體報爆滿，因此參考在[5]中透過減少特徵來實踐，

查看最佳分類器中的前 100 個加權特徵，實驗中發現了一個有趣組合。在前 100 名明顯的“影響”收費條款和功能中有許多功能具有較高的權重，但並不是人們直觀地認為是典型的影響指標，從對各個特徵的檢查中得出結論，在特定領域內，不一定建議從一個專門用於包含特別受影響的術語的資源開始。這些結果表明，與機器學習文獻中的許多其他分類任務一樣，最好從沒有人為限制的“手工製作”特徵開始。通過使用從數據導出的大特徵集，並且如果必要的話，通過特徵減少過程減少特徵的數量，可以識別數據中的相關模式，這些模式對於人類直覺可能是不明顯的。

那與我們實驗最相關的會是這個實驗[4]

其目的是找到最好的特徵工程方式來預測 IMDB 評論數據集的正面或負面評論，而我們的實驗取而代之的是預測評論 Tripstyle，但同樣追求相似的結果

	Features	# of features	frequency or presence?	NB	ME	SVM
(1)	unigrams	16165	freq.	**78.7**	N/A	72.8
(2)	unigrams	"	pres.	81.0	80.4	**82.9**
(3)	unigrams+bigrams	32330	pres.	80.6	80.8	**82.7**
(4)	bigrams	16165	pres.	77.3	**77.4**	77.1
(5)	unigrams+POS	16695	pres.	81.5	80.4	**81.9**
(6)	adjectives	2633	pres.	77.0	**77.7**	75.1
(7)	top 2633 unigrams	2633	pres.	80.3	81.0	**81.4**
(8)	unigrams+position	22430	pres.	81.0	80.1	**81.6**

另外一個在文本分類訓練最常遇到的問題也就是文本數據集的不平衡。通常有兩種處理不平衡數據集的方法，第一種通用方法是修改分類器。一些研究已經研究了成本敏感學習的使用。其中研究了成本敏感的 SVM 對文本分類的影響。[7]修改了 SVM 分類器本身以處理不平衡數據集，將問題轉變為一類分類問題。分類器在多數類上訓練，少數類則通過檢測異常來識別。

在分類器的改良和選擇上[8]設計了實驗來檢查基於成本和基於閾值的 SVM 學習算法對高度不平衡類的改進。透過控制了訓練數據中正面文檔的比例，但用自然類分佈評估了測試數據的方法。考慮了幾種修改閾值的方法。可以觀察到，雖然當正訓練樣本的數量較少時，SVM 學習器獲得的原始閾值的性能非常差，但是通過選擇不同的閾值可以顯著改善相同的模型。事實上，通過在訓練集上使用交叉驗證選擇閾值，可以達到最佳閾值的 70%，即最大化測試數據 F1 性能指標的閾值。實驗表明，儘管直接的基於成本的方法可以提高性能，但與修改閾值 b 的效果相比，這種改進的程度有限。

解決不平衡數據集問題的第二種通用方法是修改數據本身。修改數據集的最普遍和最典型的方法是重新採樣數據和對特徵選擇做變化。

重新採樣數據其中又以 Oversampling 和 Undersampling 最為常見：[10]其中有提出比較特別的方式是可以合併 Undersampling 和 Oversampling 來做採樣，處理高維度數據集由於 Naïve Bayes、kNN 和 SVM 都是流行的文本分類算法，因此在對不平衡數據集場景中的文本進行分類時，重新採樣應被視為預處理步驟。

而我們做法是對於多數類進行隨機的 Undersampling，使其跟少數類的資料量一致。

DATASET

本資料集來自於 kaggle 上 515K Hotel Reviews Data in Europe。此資料集的內容擷取自 booking.com，包含了 515,000 則客戶的評論，和全歐洲 1493 家飯店的評分。資料集總共包含了 17 個欄位，而本次實驗中主要使用的欄位分別為：Negatice_Review(負面評論)、 Postive_Review(正面評論) 和 Tag(標籤)其中本實驗的分類的 business trip 和 leisure trip 存在於 Tag 當中，而我們分別進行了文本轉換小寫和刪除 unicode 和標點符號兩項預處理。

EXPERIMENT & METHODOLOGY

1. Data cleaning

首先將 dataset 中的 Tag 欄位切出各個獨立的欄位，其中 Trip-style (內容:business/ leisure trip)，接著剔除不必要的欄位，只保留 positive/negative review 和 Trip-style 這三個欄位，最後將多數類(leisure trip)做隨機欠採樣，使得兩類資料比例一致(各 80000 多筆)，總共 16000 多筆資料

2. Preprocessing

首先先處理 missing value (Trip-style 中沒有內容的)，將 no positive/ negative 的欄位內容刪掉，這樣 word token 才不會將這些無意義的內容斷詞也做成向量，接著合併 positive/negative review 成新的欄位並轉小寫將文本做斷詞。

Word token (unigram, bigram, uni-to-bigram)

以每一個字為單位作斷詞，以利在 feature engineering 時轉成向量，清除空白或是少於五個字的評論，因為我們認字數為少於 5 個字的評論參考性價值較低。

Stop-words Removal

將 stop-words 從文本中去除，去除機器分析 feature 對文本的影響力時會混淆或參考價值低價值的原料。

Lemmatized

詞形還原(ex:將複數轉回單數形態)，在此實驗中做的是名詞的詞形還原。

2.2 Feature Engineering

2.2.1 TF-IDF

Term frequency

在一份給定的文本中，詞頻（指的是某一個給定的 term 在該檔案中出現的頻率。這個數字是對詞數（term count）的歸一

化，以防止它偏向長的檔案。（同一個詞語在長檔案裡可能會比短檔案有更高的詞數，而不管該詞語重要與否。）對於在某文本裡的詞語來說，它的重要性可表示為：

$$\mathrm{tf}_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

Inversed-document frequency
是一個詞語普遍重要性的度量。某一特定詞語的 idf，可以由總檔案數目除以包含該詞語之檔案的數目，再將得到的商取以 10 為底的對數得到：

$$\mathrm{idf}(t) = \log \frac{n}{1+\mathrm{df}(t)}.$$

Euclidean Normalize，以達成 Tf-Idf 之向量化，可得到各詞語之向量。

$$v_{norm} = \frac{v}{||v||_2} = \frac{v}{\sqrt{{v_1}^2+{v_2}^2+\cdots+{v_n}^2}}.$$

N-Gram model

是一種基於統計語言模型的算法。它的基本思想是將文本裡面的內容按照字節進行大小為 N 的滑動窗口操作，形成了長度是 N 的字節片段序列。

每一個字節片段稱為 gram，對所有 gram 的出現頻度進行統計，並且按照事先設定好的閾值進行過濾，形

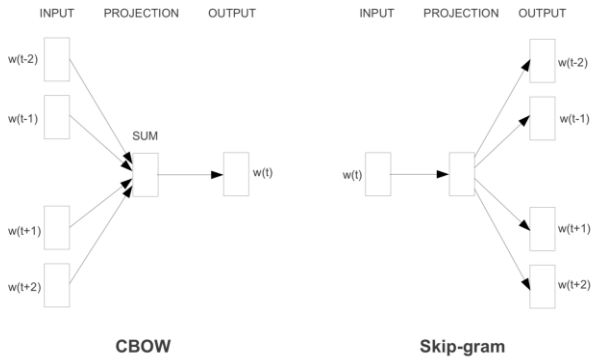
成關鍵 gram 列表，也就是這個文本的向量特徵空間，列表中的每一種 gram 就是一個特徵向量維度。

該模型基於這樣一種假設，第 N 個詞的出現只與前面 N-1 個詞相關，而與其它任何詞都不相關，整句的機率就是各個詞出現機率的乘積。

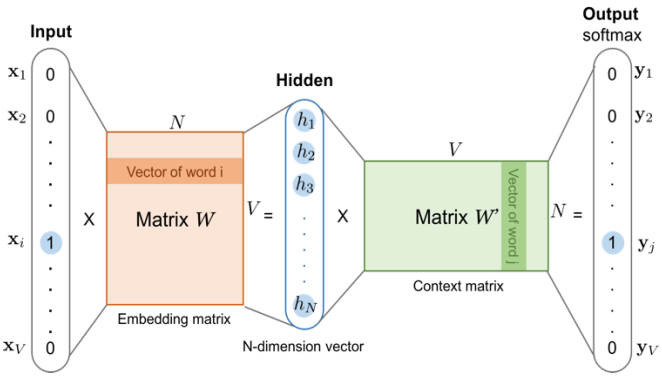
N-gram	Example
Unigram	‘to’, ‘be’, ‘or’, ’not’, ’to’…
Bigram	‘to be’, ‘be or’, ‘or not’, ‘not to’…

2.2.2 Word-embedding

- Skip-Gram 和 CBOW 模型，以目標字詞 w(t)為對象建模。設定窗口大小並投影，以建立模型。
- 神經網路現多以 Skip-Gram 為模型處理進行，建立 Full-connected 網路（fig. 2），計算隱藏層權重。將模型調整為適合類神經網路的輸入（One-Hot Encoding）和輸出層設置，隱藏層則為定義的 Hidden Layer Weight Matrix 進行計算。



- Figure 1: CBOW, Skip-gram models



- Figure 2: Skip-gram Neural Network
  - （Input：單詞做 One-hot encoding 為向量輸入。
  - Output：設定 window 參考前後文為輸出的 Target。
  - Hidden：目標權重矩陣。擷取出各文字的高維向量。）

2.2.3 Feature Selection

Doc frequency:
指的是 term 在總文本集內共出現幾次，其中如果該 term 在一篇評論有出現則為 1，沒出現則為 0，這裡只考慮是否出現，不管 term 在同一文本出現多少次都視為一次。
而在 tfidf 中我們篩選的 feature 是選出 df 值至少要 5000 以上的 term 才能作為我們的字典，並以此來建立 BOW 的向量維度。

3. Classification Model

3.1 Naive Bayes Classifier[25,26]

文字分類可視為將 Document D 分類至 Category C（此研究為 Binary classification）。訓練集中的 Instance 為（D, C），而 Document D 可經過上述項量化特徵萃取等步驟，量化得到一高維度的特徵。

可簡單表示成：

$$P(C=c|D=d) = \frac{P(C=c) \times P(D=d|C=c)}{P(D=d)}$$

3.2 Logistic regression Classifier

線性回歸是用來預測一個連續的值，邏輯回歸則是用來分類，線性回歸輸出是一個連續的實數，邏輯回歸就是用線性回歸的輸出來判斷這個資料屬不屬於 target(二分類問題: target 和 non-target)，與線性回歸概念相同，最簡單的概念，將點帶進去回歸線，回歸線輸出值若是>=0，是一類(target)，值<0 是另一類(non-target)，邏輯回歸用到的對數函數是 Sigmoid 函數。其中要如何找到 beta 參數，則可以透過 MLE 以及 Bernouli 機率分布函數來求得，公式如下:

$$\text{Target 機率: } p(y=1|\boldsymbol{x})$$

$$\text{nontarget 機率: } p(y=0|\boldsymbol{x}) = 1 - p(y=1|\boldsymbol{x})$$

$$L(\boldsymbol{\beta}) = \prod_{i=1}^{n} p_i^{y_i}(1-p_i)^{1-y_i}$$

3.3 NN

我們在此次實驗所使用的 NN 模型，是屬於比較簡單的 NN 版本，神經網路中各層的架構由兩層的 dense layer 所組成，其中 activation 函數是採用 softmax 函數。

Layer (type)	Output Shape	Param #
dense_1 (Dense)	(None, 256)	8192
dropout_1 (Dropout)	(None, 256)	0
dense_2 (Dense)	(None, 512)	131584
dropout_2 (Dropout)	(None, 512)	0
dense_3 (Dense)	(None, 1024)	525312
dropout_3 (Dropout)	(None, 1024)	0
dense_4 (Dense)	(None, 2)	2050

3.4 LSTM

我們在此次實驗所使用的 LSTM 模型，是屬於比較簡單的 LSTM 版本，神經網路中各層的架構分別由一層的 embedding layer，一層 LSTM layer 以及一層的 dense layer 所組成，其中 activation 函數是採用 sigmoid 函數。

Layer (type)	Output Shape	Param #
embedding_1 (Embedding)	(None, 30, 32)	1080576
lstm_1 (LSTM)	(None, 128)	82432
dense_1 (Dense)	(None, 2)	258

4. Evaluation

F1-Score & Accuracy

在二項分類的統計分析中， F1 score 是測試準確度的衡量方法。它考慮了測試的精準度和召回率來計算得分： precision 是正確的陽性結果的數量除以分類器返回的所有陽性結果的數量， recall 是正確的陽性結果的數量除以所有相關樣本的數量（所有樣本應該被確定為陽性）。 F1 score 是精度和召回的調和平均值 ，其中 F1 score 在 1 處達到其最佳值（完全精確度和召回率），在 0 處達到最差值，而我們在此實驗中衡量模型的標準便是看 F1-score 以及 accuracy 來判斷模型的分類的好壞。

5. Review contents rule-based classification Analysis

本專案的另一個重點，在於分析不同顧客重視之層面，我們在兩類旅遊類型中分別去探討它們的正面評論以及負面評論有提到哪些關鍵字，而我們做法是採取 rule-based 的方式建立類別，我們將顧客在旅遊住宿常見的問題定義為四類，分別為設施、服務、交通和食物，並建立四種類別之常見關鍵字，接著我們計算 term 的 df 值，並同樣保留 df 大於 5000 的 term 當作常見且為關鍵字的列表，從中一個一個分析，並將它們歸類至上數的四個類別中，以此方式完成我們自訂的 class。接著將四個類別視為四篇文章，去計算資料集中的各個評論與分別這四篇文章的 cosine similarity，藉此判斷評論之內容主要訴求的層面是屬於哪一類，最後統計 business 與 leisure 之顧客正面及負面的評論在此四種類別之佔比(注重甚麼)。我們建立的這四種類別的常見字如下:

Facility	Room, bed, comfortable, bathroom, bar, shower, view, facility, wifi, floor, window, air, door, desk, tv, décor, bedroom
Service	Staff, friendly, service, reception, check, booking
Transportation	Location, station, area, place, metro, central, city, parking, park
food	Breakfast, restaurant, food, coffee, water, tea drink

EXPERIMENT RESULT & ANALYSIS

在文字轉換成 TFIDF 的向量下，不管事使用哪種分類器，bigram 相對於 unigram 來說有稍微更好的表現，可能對於短篇評論來說，bigram 更能凸顯詞語順序的特徵，使得分類器在做分類時，更能捕捉到商業旅遊以及休閒旅遊的差別，而同時操作 unigram 和 bigram 得到 uni-to-bigram 的結果會得到略高於 bigram 的表現，同時在各個類別也都有所提高。

其實從分類器的結果我們便可知道，要區分出商業旅遊和休閒旅遊類型的評論並不容易。而其中，最好的結果也不意外是我們的 NN 神經網路模型，但比較令我們意外的是 NN 的表現也僅略高於 LR 一點點而已，由此可見，這兩種旅遊類型評論其實本質上並無太大區別。[Table 1]

Table 1

Classification model with Tfidf in min_df = 5000

model	n-gram	dim	accur	F1	Busi	Lei
NB	1	31	0.536	0.51	0.4	0.62
	2	306	0.584	0.58	0.54	0.62
	1,2	337	0.571	0.56	0.5	0.63
LR	1	31	0.57	0.57	0.55	0.59
	2	306	0.62	0.62	0.62	0.63
	1,2	337	0.621	0.62	0.62	0.63
NN	1	31	0.572	0.57	0.52	0.61
	2	306	0.62	0.62	0.61	0.63
	1,2	337	0.624	0.62	0.61	0.64

而另一種以 word-embedding 作為文字向量化的表示中，我們實作了兩種形式，第一種是 word2vec，採用一則評論中的文字向量總和取平均得到該評論的最終向量，以此代表該則評論的向量；另一種則是 doc2vec，以一則評論為單位進行 word-embedding，直接算出該評論的向量。其中 word2vec 在各分類器分類的的結果與前面的 TFIDF 表現差不多，而另一個 Doc2vec 的結果，則出乎我們意料之外，其分類的表現大大提升很多，都有達到 90%以上的成績，且不管是精準度還是召回度都是正常，沒有偏袒任何一方，但這樣的結果卻讓我們懷疑是否存在問題，這個分類問題本身應該是難以區分的，但在這樣的向量形式下，表現卻顯得如此的好，顯得有些不太合理，因此在尚未找到合理解釋的情況下，在書面報告中僅供作呈現，而我們在口頭報告中便沒有採用這樣的結果。[Table 2]

Table 2

Classification model with Word-embedding

(Window size = 2, Vector size = 300)

model	Type	accuracy	F1	Busi	lei
NB	Word2vec	0.5317	0.51	0.4	0.62
	Doc2vec	0.8954	0.9	0.9	0.89
LR	Word2vec	0.5639	0.56	0.54	0.58
	Doc2vec	0.9026	0.90	0.90	0.90
NN	Word2vec	0.5649	0.56	0.56	0.57
	Doc2vec	0.9778	0.98	0.98	0.98

接著我們同樣是採用 word-embedding 作為文字向量化的表示，但模型是採用神經網路中具有時序性的模型 LSTM 進行分類，而我們為使得每一篇評論的長度相同，我們在 max review length 的參數上有做調整，分別嘗試了 50 和 100，使得每篇評論向量長度一致，而在兩個不同的數值下，模型的表現並沒有太大差異，但在 LSTM 模型中，所跑出來的分類結果是這次所有分類模型組合中表現最好的且也是比較合理的表現，而這也是我們認為 doc2vec 相較不可信的原因，因此在這樣較難直觀判斷分類出商業旅遊以及休閒旅遊的狀況下，我們認為 LSTM 搭配 word-embedding 的組合是最佳的模型。[Table 3]

Table 3

Word-embedding in LSTM model

Model	Max-length	accur	F1	Busi	lei
LSTM	50	0.666	0.67	0.67	0.66
	100	0.667	0.67	0.67	0.66

最後一個要探討的是旅客在意飯店的四個層面的分析，經由統計的結果如下[Table 4、5]，我們可以透過這張表做出相當多的分析和解釋，這裡只簡述其中的兩種，在不管是商務旅遊還是休閒旅遊中，facility 所佔有的數字最高，代表住客最主要在意飯店的層面會是在設施的好與壞，因此正面及負面評論中都帶有相較其他三個層面最高的投票總數，而在進一步探討商務旅遊的住客給予正面回饋的多半是交通這方面，在生活經驗中也屬合理，因為商務旅遊最在意的當然還是交通方便能夠與商務行程上的安排的便利性因此當飯店在淡季要服務較多商務旅遊的住客時，我們便會建議該飯店可以針對交通上做考量，提供接駁等可以提供住客交通便利的措施，藉此提高滿意度；如果是探討休閒旅遊的住客所在意的層面的話，服務方面是他們給予較多正面回饋的，在生活經驗上也確實合理，今天在休閒方式去旅遊的情況下，飯店服務的好與壞，也間接了影響旅客此趟旅遊的心情，因此當飯店在旺季要服務較多休閒旅遊的旅客時，我們便會建議該飯店可以加強服務上面的品質，甚至在服務上給予優惠等等措施，藉此來建立口碑。

Table 4、5

four level of customer concerns

	facility	service	transport	food
Neg_business	26821	8144	4278	9740
Pos_business	16846	18810	19565	9630

	facility	service	transport	food
Neg_leisure	24772	6797	3652	10216
Pos_leisure	19876	22219	18381	9547

DISCUSSION

1. Feature selection 的方式可以探討更多方式去做處理，像是上課所教的 log likelihood ratio, chi-square 方法等等，選出來的特徵可能會不同，效果可能也會較好也說不定。

2. 由於記憶體問題，我們將龐大 dataset 經 preprocess 完及隨機欠採樣後的結果可能或多或少刪除了評論的重要特徵，或許在解決數據量太大跑不動的問題後，更能發揮龐大 dataset 中這麼多可參考資料的最大價值。

3. 修改分類器，可嘗試多種其他適合文本分類的分類器(如 SVM)，修改分類器(threshold)值，或是用別的分類器來決定哪些 feature 是否該被參考，進行 feature reduction。

4. 參數該如何調整(min_df)，目前有調整過後的跑的結果反而較好，可多調整幾次找出跑出來效果更好的參數。

5. Unseen words 的處理方式: 該如何給予向量，可以考慮做 smoothing。

6. 由於我們在神經網路模型上只使用最簡單的版本來跑，或許在多加了幾層 hidden layer 做訓練之後跑出來的效果會有所突破。

CONCLUSION

以此次實驗結果來說，最好的分類模型是doc2vec搭配NN模型以及word-embedding搭配LSTM，相較於Word Embedding的算法，Bag Of Word所取出來的feature鑑別力確實也比較差，而這樣的實驗結果，對於神經網路模型跑出來的效果比機器學習好也與我們當初預期的結果相同，但相較於後者，前者doc2vec在各分類器跑出來的效果竟如此地高，甚至是超越神經網路模型，且在precision與recall值都表現正常無任何偏頗的情況下，我們認為有些不太合理，因此我們最後仍是以後者的LSTM模型為主，畢竟在神經網路的模型的實驗結果下，以及人工去看兩種旅遊類型不同的評論差異，確實很難分辨，因此最好的分類模型我們認為會是word-embedding搭配LSTM。對於資料集的不平衡問題，我們選擇透過隨機under-sampling來進行對應的處理，但其實驗結果在Bag of Word和Word Embedding中並沒有特別高的分數表現，我們認為進行under-sampling後可能也相對應地遺失了一些可用來判斷的文字的特徵，如果能在保全其所有資料來做分類，可能會使得本實驗專案的內部效度更高。預測旅客旅遊類型可推出適當的促銷活動、分析不同客戶評論之內容可協助飯店進行優化帶來更多效益，而我們所分析出不同旅遊類型的旅客注重飯店不同層面的結果也確實可以反映顧客所在乎飯店各層面的比例，飯店方也可依據這樣的分析結果在淡旺季的時候制定相對應的經營策略做改善，另一方面對於其他無法得知住客的旅遊類型的訂房服務平台來說，透過我們實驗專案模型的雛形，可以去往下研究並製作即時的線上評論分辨旅遊類型的模型，了解哪種飯店會受什麼樣旅遊類型的住客青睞之後、推出更準確且即時性的推薦系統。

MEMBER'S WORKLOAD

整個專案皆由三人合力構思所完成

成員	負責內容
余知諺	code(資料集預處理、特徵工程、分類模型)、書面報告
陳映樵	code(顧客注重飯店的不同層面分析)
黃思凱	Presentation & ppt 製作

REFERENCES

Paper or experiment related to Review

[0]Quoc Le QVL,Tomas Mikolov "Distributed Representations of Sentences and Documents" Google Inc, 1600 Amphitheatre Parkway, Mountain View, CA 94043

Text Feature extraction method

[1] Arjun Mukherjee and Bing Liu Department of Computer Science University of Illinois at Chicago, IL 60607, USA "Aspect Extraction through Semi-Supervised Modeling"

[2] Turney, P.D.: "Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews." In: Procs. of ACL. (2002) 417–424

[3] Riloff, E., Wiebe, J., Wilson, T.: "Learning Subjective Nouns Using Extraction Pattern Bootstrapping." In: Procs. of CoNLL. (2003) 25–32s

[4] Pang, B, L.L., Vaithyanathan, S.: "Thumbs up? sentiment classification using machine learning techniques." In: Procs. of EMNLP. (2002) 79–86

[5] Gamon, M.: "Sentiment classification on customer feedback data: Noisy data, large feature vectors and the role of linguistic analysis." In: Procs. of COLING. (2004) 841–847

Dealing with imbalanced text dataset

[6] Brank J., Grobelnik M., Milic-Frayling N. & Mladenic D. (2003) "Training text classifiers with SVM on very few positive examples." Report MSR-TR-2003-34

[7] Liu A. Y. C. (2004) "The effect of oversampling and undersampling on classifying imbalanced text datasets." Masters thesis. University of Texas at Austin

[8] Zheng Z., Wu X. & Srihari R. (2004) "Feature selection for text categorization on imbalanced data." ACM SIGKDD Explorations Newsletter: Special issue on learning from imbalanced datasets 6:80–89

Text classification 的相關實驗

[9] Yao, H., Liu, C., Zhang, P. “A feature selection method based on synonym merging in text classification system” et al. J Wireless Com Network (2017) 2017: 166.

Online resources

[10]Re-sampling Imbalanced Training Corpus for Sentiment Analysis
https://medium.com/@muabusalah/re-sampling-imbalanced-training-corpus-for-sentiment-analysis-c9dc97f9eae1

[11]Feature Extraction from Text
https://andhint.github.io/machine-learning/nlp/Feature-Extraction-From-Text/

[12]自然語言處理入門- Word2vec 小實作
https://medium.com/pyladies-taiwan/%E8%87%AA%E7%84%B6%E8%AA%9E%E8%A8%80%E8%99%95%E7%90%86%E5%85%A5%E9%96%80-word2vec%E5%B0%8F%E5%AF%A6%E4%BD%9C-f8832d9677c8

[13]Doc2vec
https://medium.com/@mishra.thedeepak/doc2vec-simple-implementation-example-df2afbbfbad5

[14]LDA
https://taweihuang.hpd.io/2019/01/10/topic-modeling-lda/

[15]N-gram (在 preprocessing 做，再做 tfidf 轉向量，可嘗試合併多個不同 N-gram 向量)
https://medium.com/machine-learning-intuition/document-classification-part-2-text-processing-eaa26d16c719

[16] Traditional Methods for Text Data
https://towardsdatascience.com/understanding-feature-engineering-part-3-traditional-methods-for-text-data-f6f7d70acd41

[17] Understanding difference between business and leisure trip
https://www.emarketingassociates.com/blog/understanding-differences-business-vs-leisure-travelers

[18] Simplifying the rift between leisure and business travel
https://www.e-marketingassociates.com/blog/simplifying-the-rift-between-leisure-and-business-travel

[19] Unsupervised Approaches for Automatic
Keyword Extraction Using Meeting Transcript

[20] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. In ICLR.

[21] Feature Selection for Text Categorization on Imbalanced Data

[22] The Optimality of Naive Bayes

[23] Combining Naive Bayes and n-Gram Language Models for Text Classification (2003)

[24] Word2Vec Tutorial - The Skip-Gram Model
http://mccormickml.com/2016/04/19/word2vec-tutorial-the-skip-gram-model/

[25] N-gram
https://read01.com/M2mJB63.html

[26] Logistic Regression
https://medium.com/@chih.sheng.huang