法規資料搜尋引擎

組員:10520136 黃少麒
10520121 杜欣洋
10520104 莊東翰

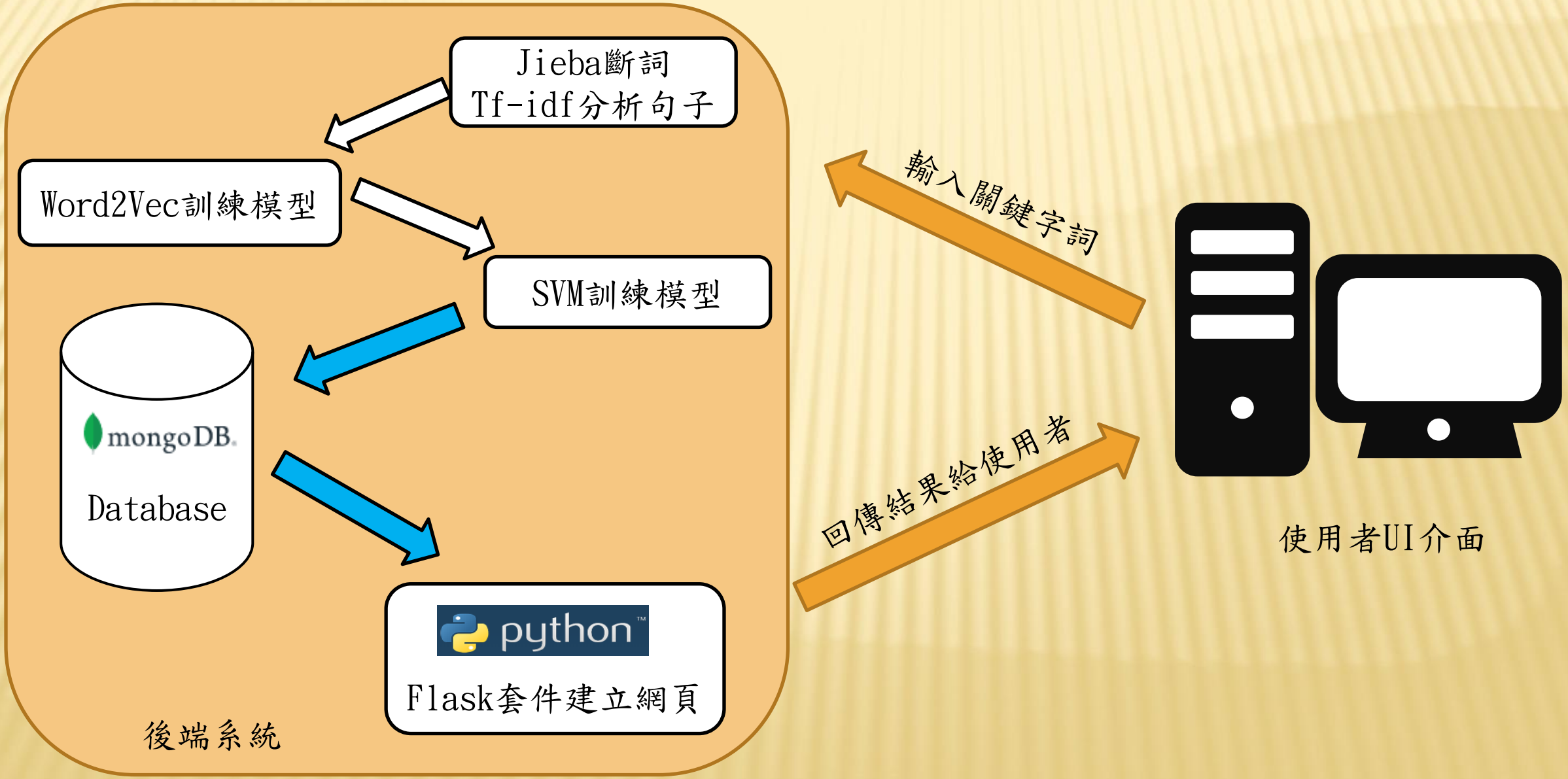
指導老師:吳宜鴻老師

OUTLINE

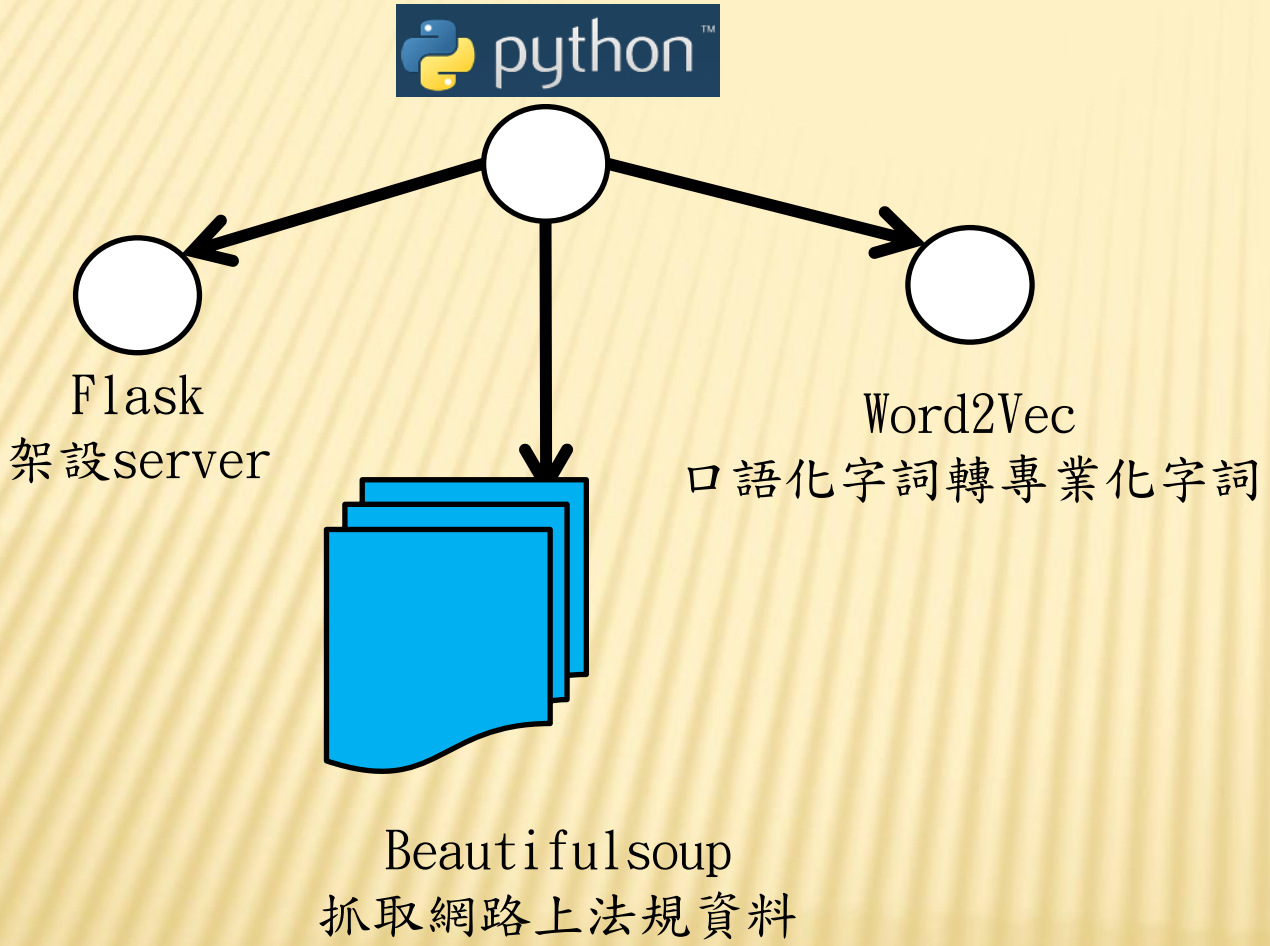
- 動機與目的
- 系統架構
- 使用工具
- 分析方法
- 結果呈現
- 問題與未來展望

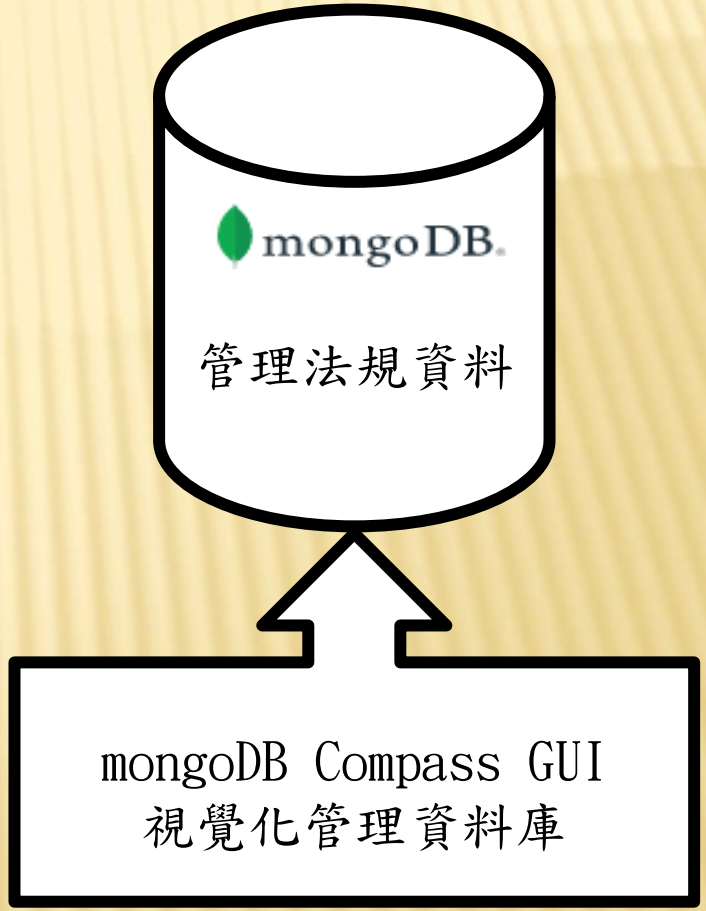
動機與目的

系統架構


Jieba斷詞
Tf-idf分析句子
Word2Vec訓練模型
SVM訓練模型
mongoDB.
Database
python
Flask套件建立網頁
後端系統
輸入關鍵字詞
回傳結果給使用者
使用者UI介面

使用工具


python
Flask
架設server
Word2Vec
口語化字詞轉專業化字詞
Beautifulsoup
抓取網路上法規資料


mongoDB.
管理法規資料
mongoDB Compass GUI
視覺化管理資料庫

分析方法

1.Jieba斷詞

2.TF-IDF

3.SVM (Support Vector Machine 支援向量機)

4.Word2vec

Jieba斷詞

如何從句子中取出詞彙

1. Jieba斷詞:

關鍵字：連續殺人犯到底判什麼刑

連續　殺人犯　到底　判　什麼　刑

2. 刪除停詞: 因為、所以、什麼、到底…

獲得結果為：[連續，殺人犯]

TF-IDF

TF-IDF: 將文字轉成數字數據，利於分析。

TF:詞-單一句子, IDF:詞-整體性, TF-IDF:詞-兩者皆考慮

句子1:連續殺人犯判什麼刑 ⟶ “連續”, “殺人犯

句子2:殺人犯有追訴權時效嗎 ⟶ “殺人犯”, “追訴”, “時效”

單一句子: TF(“殺人犯”,句子1) = 1(出現“殺人犯”次數)/ 2(斷詞數) = 0.5

整體: IDF(“殺人犯”) = log(總共幾個句子/ 幾個句子出現“殺人犯”) = log(2/2) = 0
(表示“殺人犯”在這兩句子中沒有指標性)

單一句子: TF-IDF (單位: 一個句子) = TF(單位: 一個句子) x IDF (單位: 一個數字，表整體數值)

(同時考慮一個字在一個句子裡的占比 & 這個字在所有句子中是否具指標性)

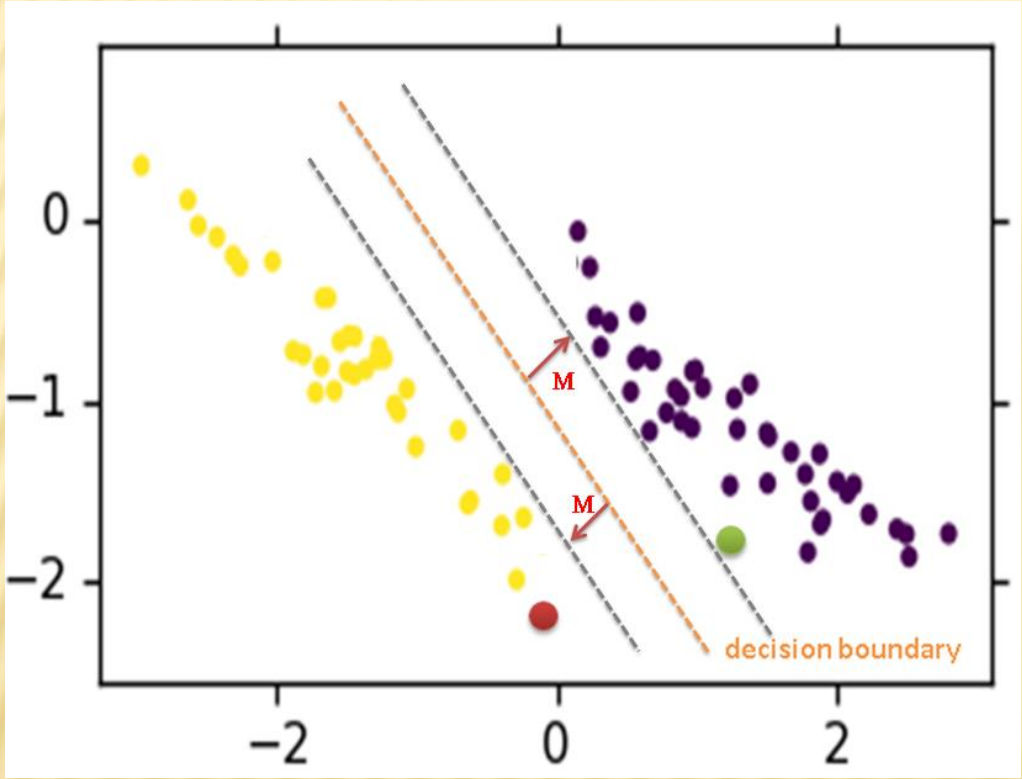
事前訓練模型

- 事先訓練好的內容：SVM訓練集、Word2vec
- 訓練模型的好處：把模型存下來後不用重複訓練，呼叫取用即可得到結果。
- SVM訓練集(Support Vector Machine 支援向量機)：尋找兩組dataset的邊界線
- Word2vec: 尋找詞的相關性，口語化的詞轉換為專業用語的步驟

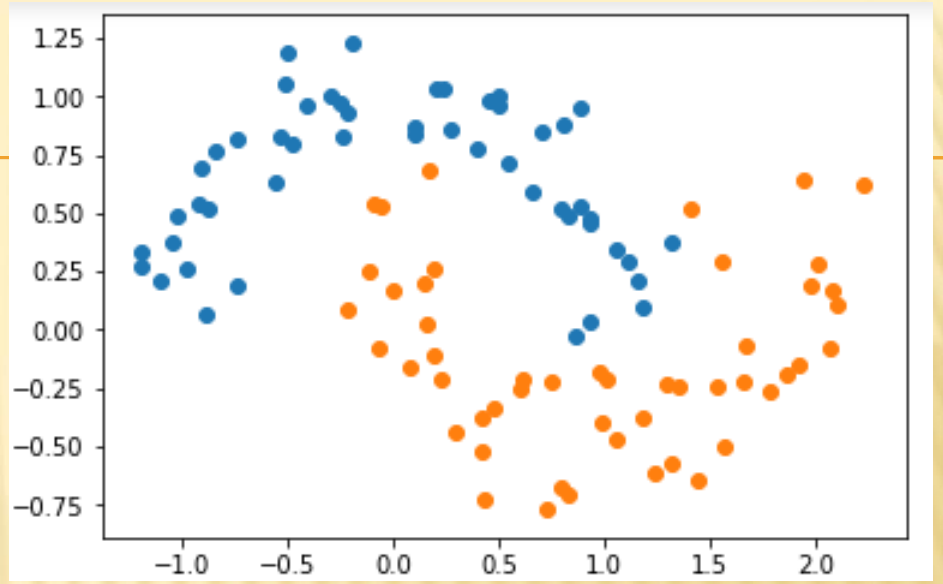
SVM訓練模型(1)

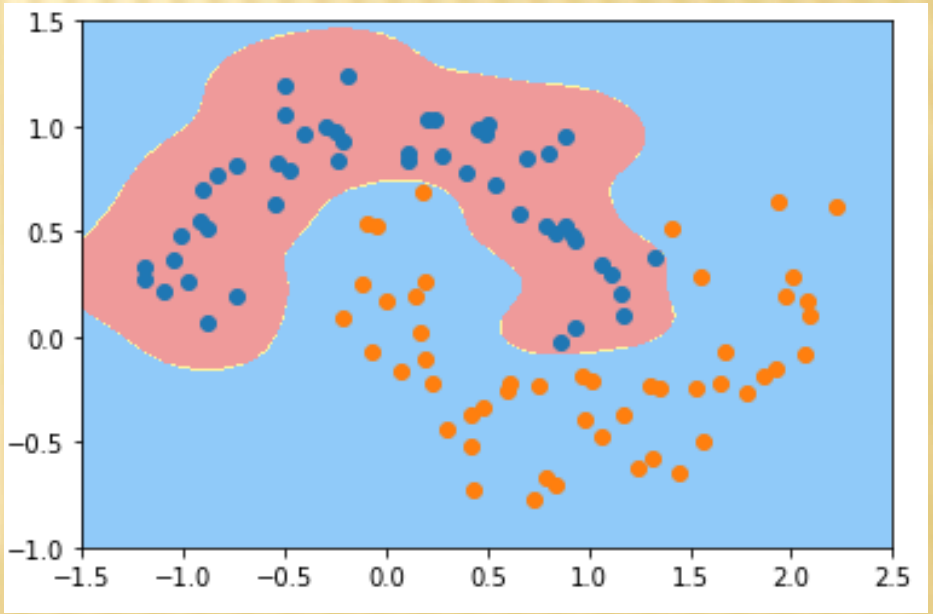
SVM (Support Vector Machine 支援向量機):

將兩組不同的dataset能用一條邊界線分開，適合用來做二元分類器(binary classifier)。



Linearly Separable





Kernel trick:

2D甚至多維空間無法用直線分割的圖形

SVM訓練模型(2)

假設SVM訓練集中只有這兩句子，欲求這兩句子在車禍問題分類中的分數:

句子1: 在車禍中受傷，可以向肇事者請求賠償因為受傷減少的收入嗎？
“車禍”，“受傷”，“肇事者”，“請求”，“賠償”，“受傷”，“減少”，

句子2: 遇到車禍受傷的時候，我可以在刑事訴訟中也請求賠償金嗎？
“遇到”，“車禍”，“受傷”，“刑事訴訟”，“請求”，“賠償金”

交通

交通規則　車禍問題

此為句子1的其中一個feature。共2種分類，所以此例每個句子會有2個feature。

	車禍	肇事
句子1	1/8 = 0.125	1/8 = 0.125
句子2	1/5 = 0.2	0/5 = 0

TF

x

	IDF
車禍	log(2/2) = 0
肇事	log(2/1) = 0.301

=

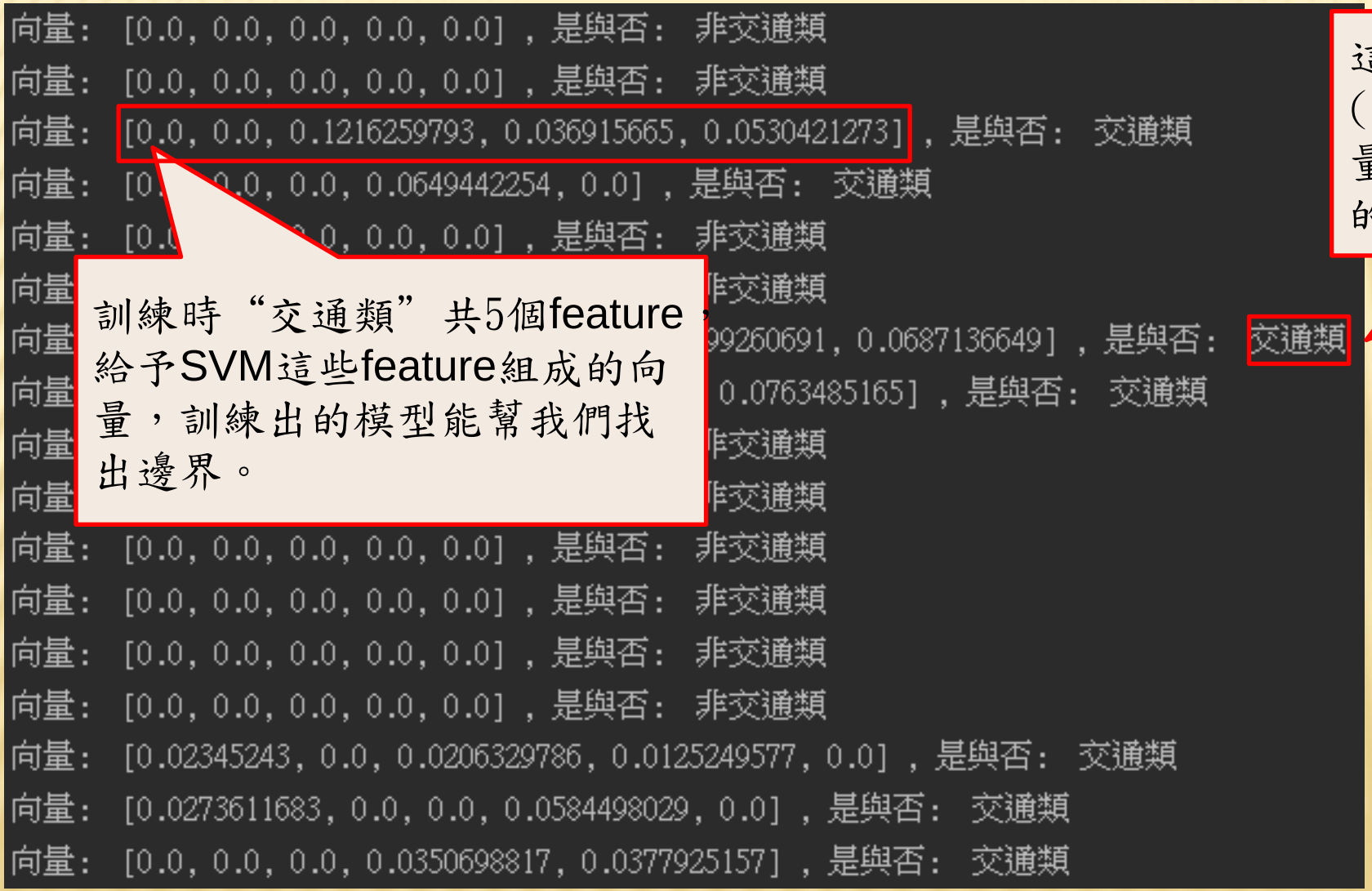
	車禍	肇事
句子1	0.125*0 = 0	0.125*0.301 = 0.037
句子2	0.2*0 = 0	0*0.301 = 0

TF-IDF

	此類別的feature
句子1	0+0.037 = 0.037
句子2	0+0 = 0

實際上不可能為0，因為不可能訓練集中每一句子都有“車禍”

SVM訓練模型(3)



訓練時“交通類”共5個feature，給予SVM這些feature組成的向量，訓練出的模型能幫我們找出邊界。

這些分類在訓練時由人給予(已識別過)，由人判讀的少量訓練集達成半自動訓練目的。

SVM訓練模型(4)

若今天來了一筆新的向量(使用者問題)，就能得到結果

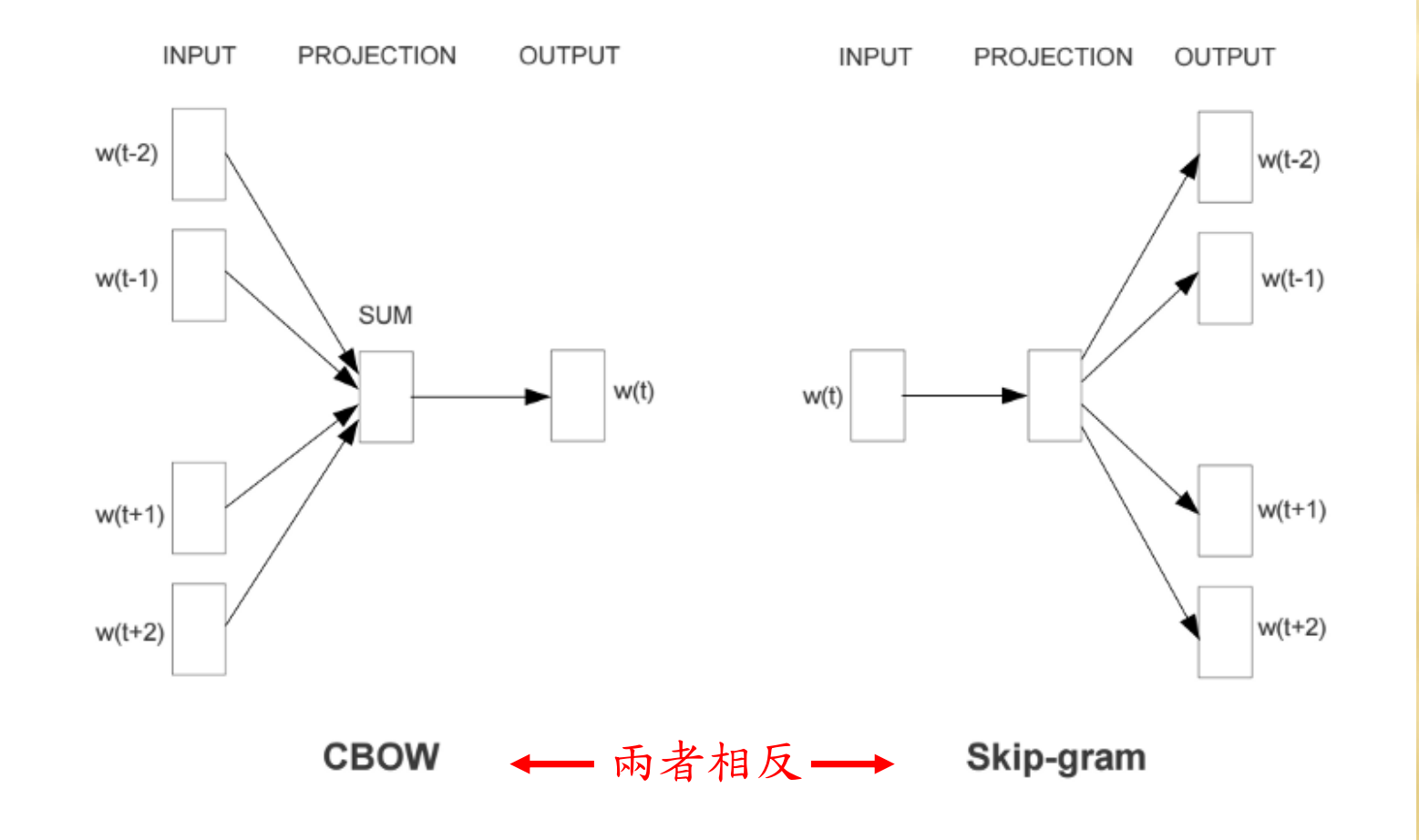
```
使用者的問題: 闖紅燈發生車禍，有肇責嗎?
```

```
print("使用者問題轉換成向量: ", YourQ)
pred = model.predict(YourQ)
print("結果: ", pred)
```

```
使用者問題轉換成向量:  [[0.3283340198, 0.2855973665, 0.1444308505, 0.1753494085, 0]]
結果:  ['交通類']
```

Word2vec(1)

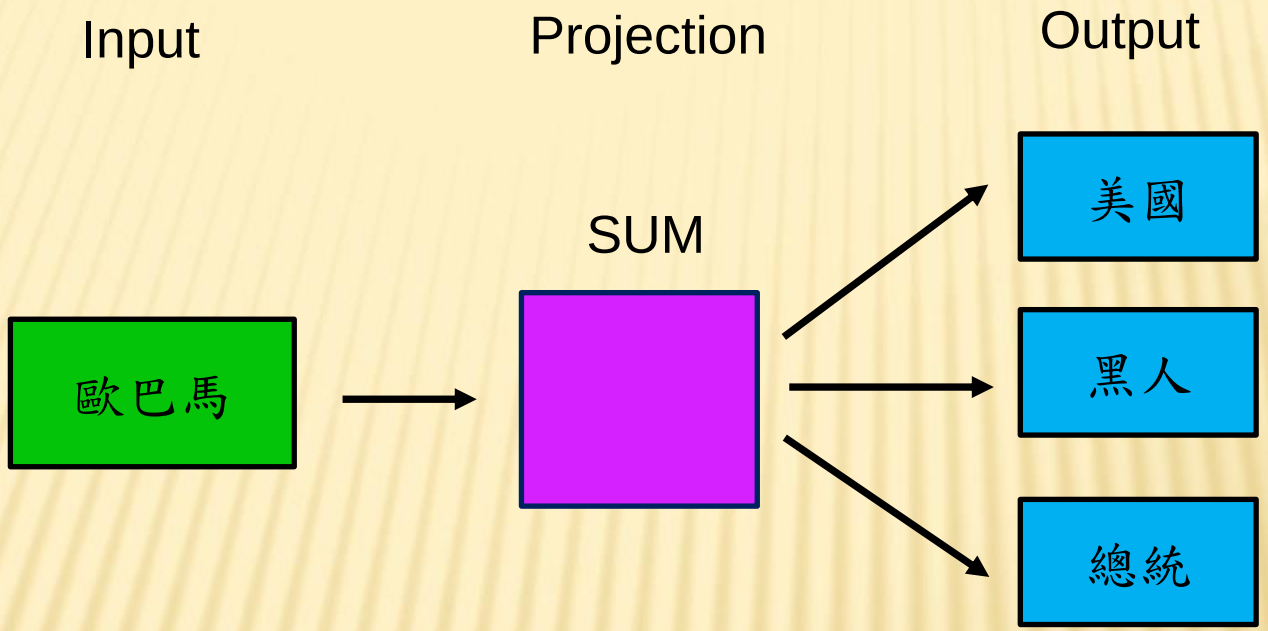
Word2vec: 入門NLP (Natural Language Processing,自然語言處理)方法



CBOW: 利用上下文來預測中心詞

Skip-gram: 利用中心詞來預測上下文

Word2vec(2)



Skip-gram

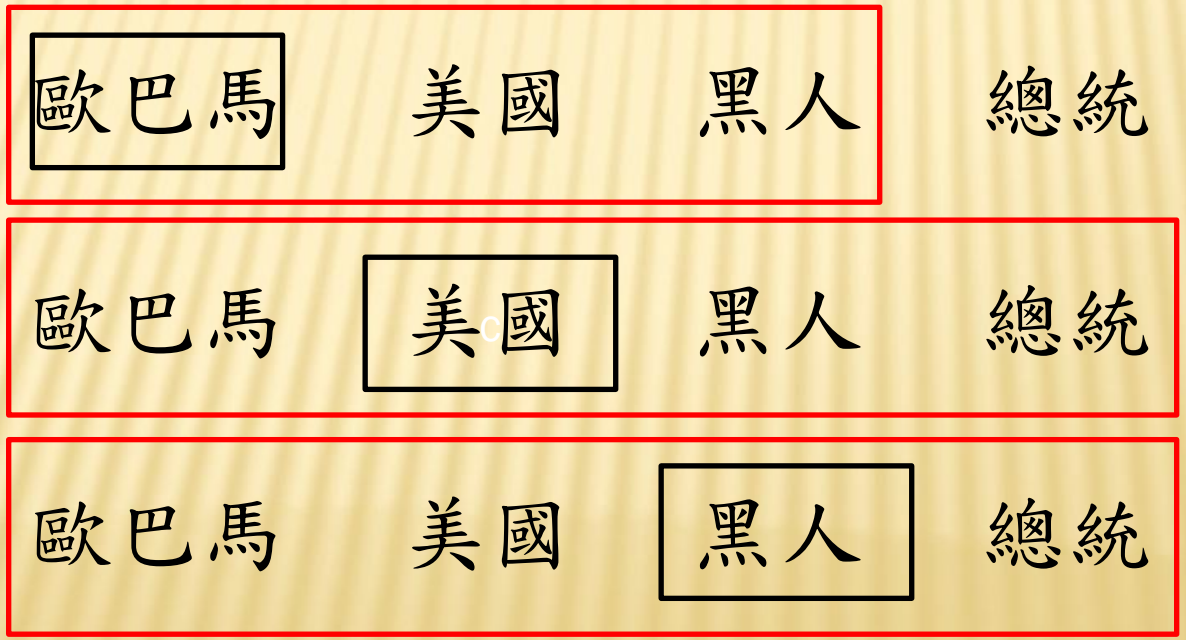
Word2vec(3)

歐巴馬是美國的黑人總統

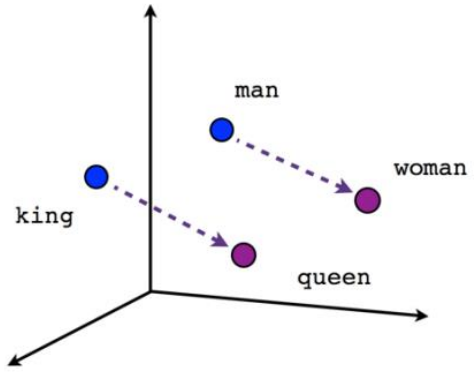
Jieba斷詞後: 歐巴馬　是　美國　的　黑人　總統

去除停詞後: 歐巴馬　是　美國　的　黑人　總統

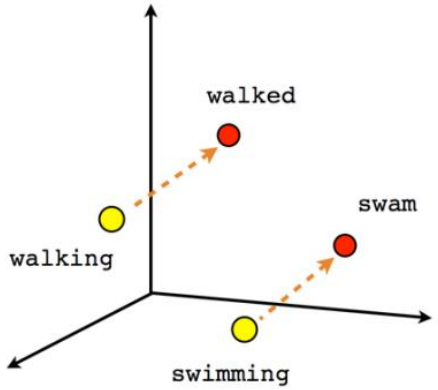
Window size = 2



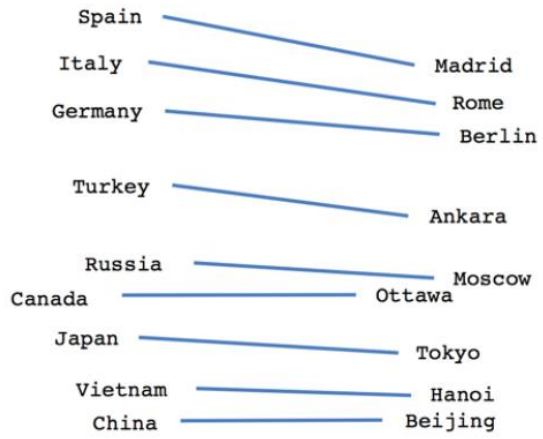
Word2vec(4) 口語化



Male-Female



Verb tense

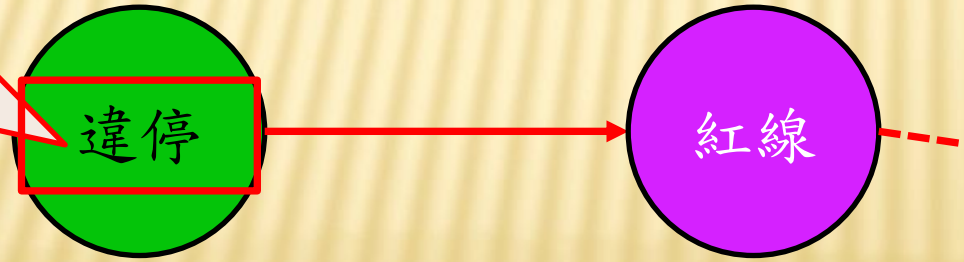


Country-Capital

違停的相關詞

```
排名 1   ('臨停', 0.9507654309272766)
排名 2   ('黃線', 0.9269193410873413)
排名 3   ('停靠', 0.9211543202400208)
排名 4   ('路邊', 0.9199017286300659)
排名 5   ('紅線', 0.9194145202636719)
排名 6   ('車在', 0.9020031690597534)
排名 7   ('警示燈', 0.8993890881538391)
排名 8   ('駛出', 0.8963791131973267)
排名 9   ('停好', 0.8948168754577637)
排名 10   ('兩輛', 0.8933905363082886)
```

“違停”原先不屬於“交通”類別中詞彙，卻能得知彼此相關。



（違規停車）
口語化詞彙

專業化詞彙

交通相關詞彙
鐵路
黃線
紅線
高速公路
國道

結果呈現

↑在我們的系統中輸入“賣國”可以一目了然的找到所有可能相關的法規。

↓在全國法規資料庫輸入賣國出現的是許多的法規名稱需逐一確認才可以找到需要的資訊。法規進入卻發現與想要的資料不符。

問題

- 小分類的判定不夠精準
- 搜尋效率可以再提升(3秒->立即顯示)
- 結果排序分數有待加強

未來展望

- 找到新的方法判斷使用者搜尋之語意
- 擴充可搜尋之法規類型
- 增加斷詞準確度
- 強化口語化搜尋之準確率

資料來源:

1. https://www.itread01.com/content/1534167680.html
2. https://chtseng.wordpress.com/2017/02/04/support-vector-machines-支援向量機
3. https://medium.com/@tengyuanchang/讓電腦聽懂人話-理解-nlp-重要技術-word2vec-的-skip-gram-模型-73d0239ad698/
4. https://leemeng.tw/find-word-semantic-by-using-word2vec-in-tensorflow.html