一、關聯分析背景

由於文本數量過大，經刪去停用詞後依然有著近 88K 的詞，嘗試幾次都會爆 RAM，因此此分析只包括部分辭彙關聯結果，

發現文本中並非全都是長篇、前後文連貫的文本，因此選擇不用 sentence transformer，反而有些是片段的名詞、以及敘事句，因此我將此篇關聯分析的範疇定義為*關鍵字分析*。

也因為專注於關鍵字分析，我的分析步驟在斷詞前先把金融關鍵詞 load 進 jieba 的 user dict，以免斷詞將有意義之金融詞斷開。後再把其餘停用詞及噪聲去除得到 processed_doc。

接下來使用 TF-IDF 取得關鍵詞，考量過於低頻過高頻辭彙都並非決定一個文本類別的關鍵詞特性，因而 TfidfVectorizer 中採用 min_df=50, max_df=500(意即最少出現 50 文本中最多出現 500 文本中)，來取得這個區間的關鍵詞(adjusted_vocab)共 3733 個，後續分析便是基於此 adjusted-開頭的變數進行分析。

由於此機制是 reusable mechanism，下方關聯結果便只採 1 關鍵字以及 1 文本做關聯性分析。

二、關聯結果

找個關鍵字/文本、結果

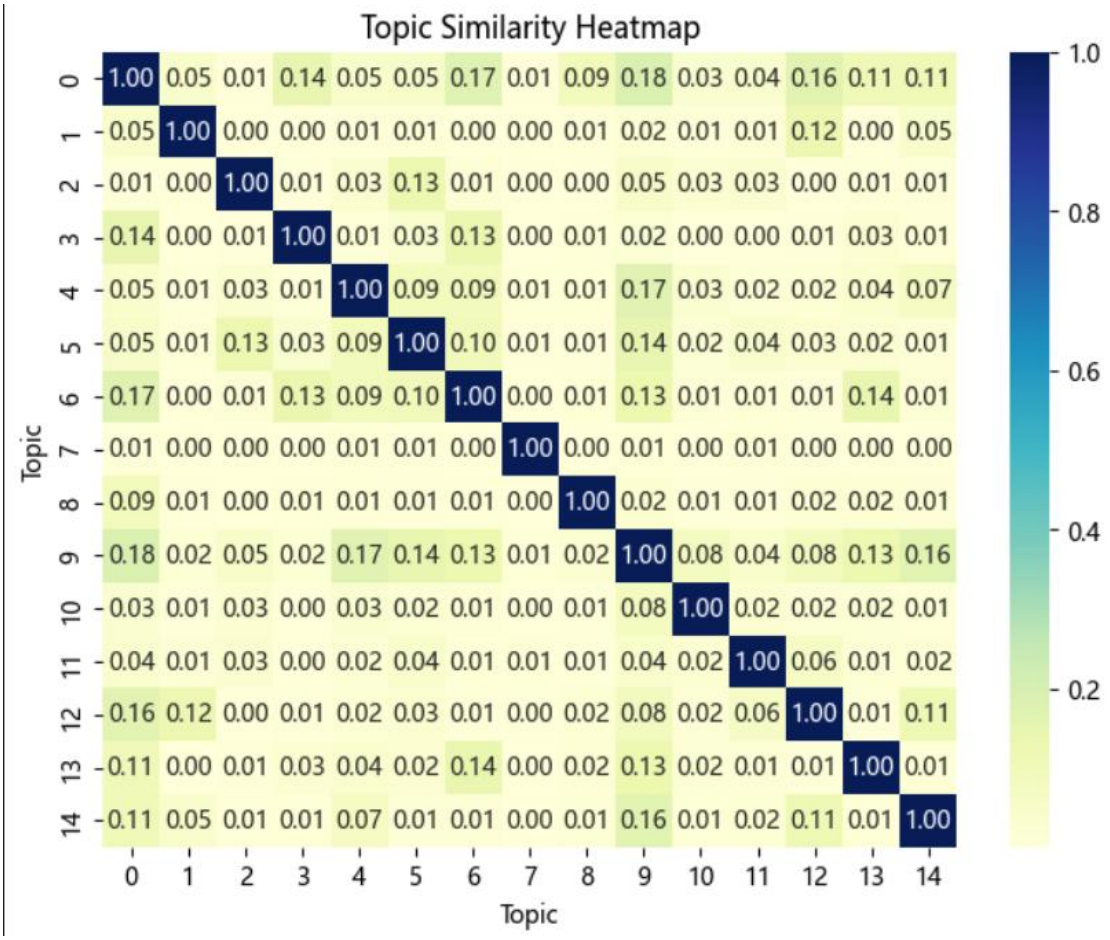
(Non-negative Matrix Factorization)NMF 主題分布結果

```
主題 0: 幾何, 巴黎, 代數, 高等, 論文, 約翰, 哲學, 學會, 猜想, 皇家
主題 1: frac, pi, sqrt, sum, 公式, sin, cos, dx, cdot, 方程
主題 2: gdp, 人均, 購買力平價, 本幣, 比重, 總值, 位次, 人民幣, 全省, 推算
主題 3: 中國科學院, 院士, 當選為, 物理系, 北京大學, 所長, 部委, 學部, 曾任, 清華大學
主題 4: 證券, 控股, 股票, 收購, 業務, 資產, 上市, 億元, 旗下, 港元
主題 5: 公裡, 開發區, 街道, 平方, 經濟技術開發區, 鐵路, 區劃, 規劃, 天津, 億元
主題 6: 行政院, 委員, 國立, 經濟部, 董事長, 部長, 國民黨, 國民, 院長, 出任
主題 7: 本分, 模板, 歸納, 部位, 收錄, 總部, 分類, 總部設, 用戶, 商場
主題 8: of, dat, name, may, 斯特, der, gust, july, 猜想, au
主題 9: 農業, 勞動, 土地, 資本, 出口, 需求, 石油, 資本主義, 社會主義, 通貨
主題 10: 背面, 正面, 硬幣, 紙幣, 流通, 面值, url, 歐元, iso, 圖案
主題 11: nbsp, hans, hant, 擴充, 透過, 空間, font, 序列, cite, 區劃
主題 12: 集合, 空間, 元素, 整數, 符號, 定義, 實數, 子集, 序列, 演算法
主題 13: term, 大臣, 總統, 總理, 民主, 議員, 內閣, 選舉, 部長, 眾議院
主題 14: 模型, 變數, 隨機, 機率, 利率, 風險, 假設, 預測, 組合, 資產
```

由主題分類可以大略看出不同主題間明顯的區別，有了這樣的主題分群步驟比較有利於後面文本間的分析，並且可以利用 topic_term_matrix(如下圖主題 0 為例)查看該主題權重最高的關鍵字，進而確認該主題對於 top K 關鍵字權重來確認主題，如主題 0 為古典數學理論、主題 2 便為總體經濟、主題 4 為證券市場等文本。

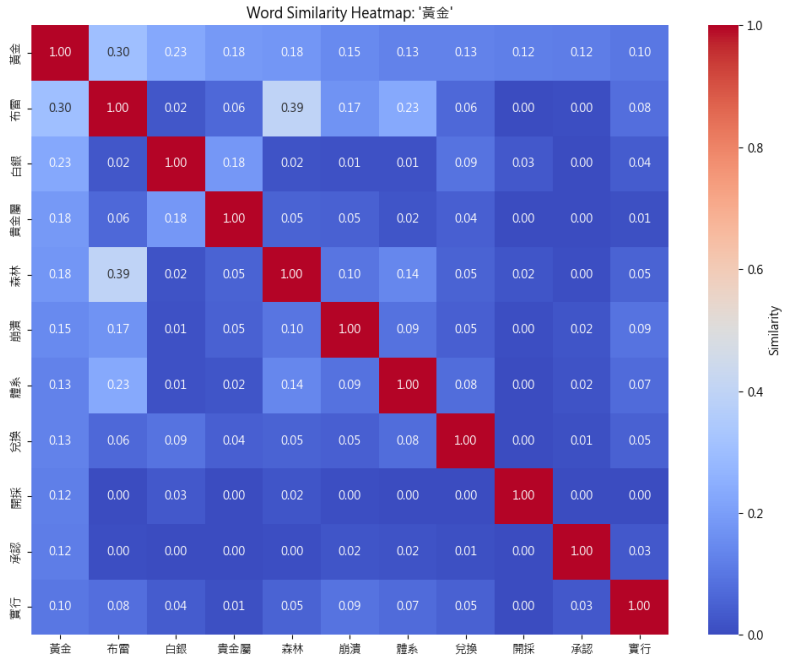
	Weight
幾何	1.440791
巴黎	1.062901
代數	0.952459
高等	0.783390
論文	0.699027
約翰	0.695768
哲學	0.682542
學會	0.667340
猜想	0.662873
皇家	0.639505
科學院	0.617665
哈佛大學	0.590423
任教	0.582051
天文	0.579055
普林斯頓大學	0.537821
數論	0.495525
劍橋大學	0.482097
院士	0.473570
數學系	0.473356
學生	0.466319

並且進一步查看主題相似度矩陣如下圖，發覺主題間顏色偏淺色，因此各主題間都不具備高重複性或高相似度，可推論主題分群成功。



詞關聯分析

基於 Appendix 所指出此資料分析涵蓋範圍及可用關鍵詞，我在此選擇「黃金」作為本關鍵字相似度分析，分析結果如下：



根據網路資訊，文本中的布雷指的是布列敦森林體系，又稱布雷體系，該制度核心便是美元與黃金掛勾，因此可見相似字中「布雷、森林、崩潰、體系、兌換、承認、實行」等關鍵字為黃金最相似的關鍵字極其合理，且都指向該體系的瓦解。

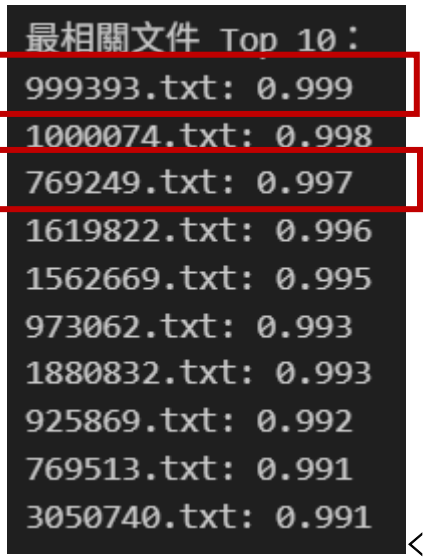
另外一組我視為「白銀、貴金屬、開採」，此組便是單純的講解黃金與白銀的刺場屬性極其相關。

整體而言 adjusted 過後的 TF-IDF 矩陣經過 normalization 後，其相似度可以說是表現不錯。

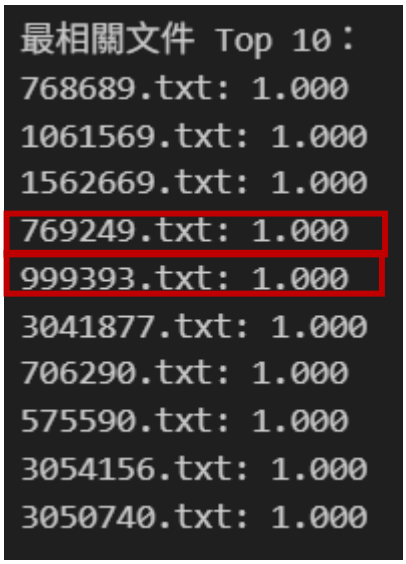
文本分析 -- 僅基於 adjusted_tdidf 矩陣比對相似度 v.s. 基於 NMF 主題模型分析結果

此部分比較了兩種分析視角，*第一種單純利用 adjusted_tfidf 取得相似文本，並做後續分析。*

第二種是基於 NMF 主題模型，該方法是透過 adjusted_tfidf 矩陣 (n_doc*n_doc)將文本分群到 15 個主題後(n_doc*n_topic)，再 normalization(n_doc*n_topic)後取 cosine_similarity(n_doc*n_doc)，利用主題性而得的相似度矩陣。



< 基於 adjusted_tdidf 矩陣比對相似度



基於 NMF 主題模型相似度矩陣 >

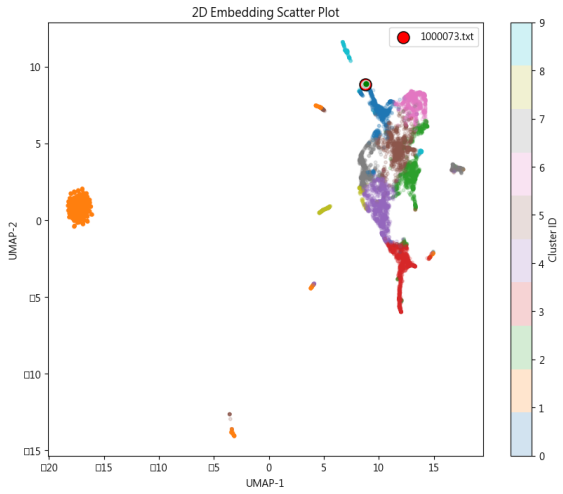
單從兩者的相似度數值比較並取 Top 10 可發現有兩檔案同時出現在這兩方法中，因此可以證明即便將 TF-IDF 轉變較為泛化的相似度矩陣仍然保有部分原始相似度。(這邊旨在證明利用 NMF 進行較為泛化的處理之後不會令資料完全失真，而且能夠以更為淺顯易懂的方法進行相似度比較)

< *基於 adjusted_tdidf 矩陣*

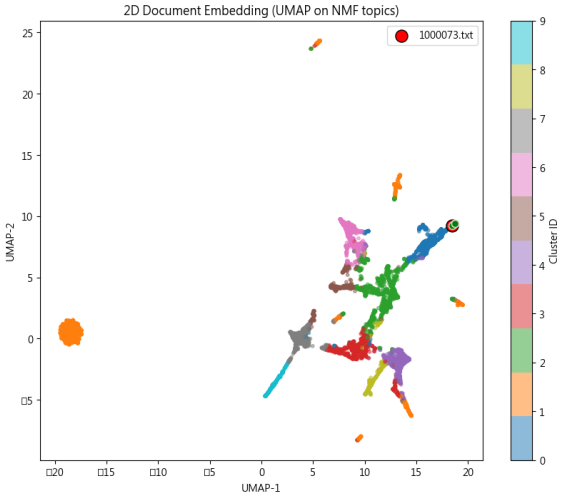
基於 NMF 主題模型 >

該組關鍵字比對呈現出同一文本不同分析模型下取得的關鍵字結果，單純用 adjusted_tfidf 選出的關鍵字較為有意義，而基於主題模型的則選出一些未清理乾淨的文字，且權重很高，顯示出該主題可能都是些有相似狀況未被清理乾淨的文檔。

但兩種分析方法仍然展現出 2 個關鍵詞的重疊，且 cluster 相同，可見該文本應該是在講區域開發有關的主題。



< 基於 adjusted_tdidf 矩陣



基於 NMF 主題模型 >

從 scatter plot 可以看出文件分群狀況極為相似，並且目標文本與相似文本所在 cluster(紅點以及綠點)都在藍色 cluster 中，即代表該文本就是屬於該 cluster。

三、演算設計

設計架構

原始 initial_tfidf (n_docs * n_words) (9943 * 726031)維度爆炸，進行清理與降維

➔ 去除停用詞 cleaned_tfidf (9943 * 87959)維度依然高，因此考慮減少 n_words 數

➔ 只取 min_df=50~max_df=500 的字 adjusted_tfidf (9943 * 3733) 維度可

接受，進行主題分類並進一步將 adjusted_tfidf matrix L2 正規化以利用 cosine similarity 計算詞相似矩陣 word_sim_matrix

➔ 基於 adjusted_tfidf，使用 NMF 將主題分 15 群，並且主題關鍵字具備可解釋性，接著也將文本主題矩陣 L2 正規化並計算 cosine similarity 得到基於主題分布的 doc_sim_matrix。(選用 NMF 是因為可提供可解釋的 topic-term 權重，利於人工審閱主題與提取關鍵詞，且由於採用 Tf-idf，必是非負矩陣，因 NMF 能夠作為快速的分類主題工具。)

➔ 將此做為後續分析主要框架，後續接續使用 adjusted_tfidf 及 NMF 分群結果

➔ 設計詞關聯 pipeline

- get_similar_words(keyword, top_k)：在 adjusted_vocab 中找 index，回傳 word_sim_matrix 排序 top_k。
- 使用者輸入關鍵詞，可直觀查看該詞在 adjusted_vocab 語料庫裡的相關詞與關聯程度與視覺化

➔ 設計文本關連 pipeline

文本關連分析有兩 pipeline，但僅有傳入的資料差距，步驟上都一樣，因此下面說明統一步驟

- 因為原始矩陣維度太高，先把高維度的 Adjusted TF-IDF 矩陣壓縮到 15 維，得到 doc_embeddings
- 向量長度調整為 1，這樣在計算 cosine similarity 時，結果只反映方向
- get_similar_docs 找出與目標文件相似的 top_k
- extract_keywords 從矩陣中找出該文件 Top K 關鍵詞
- cluster_docs 用文件嵌入向量做分群
- plot_doc_similarity → 顯示相似度分布。
- plot_network_graph → 顯示文件之間的網路關係。
- plot_document_scatter → 顯示文件在降維空間的分布，並標出 target 與相似文件

四、Appendix

資料分析涵蓋範圍 與 關鍵字示例:

```
['1000073.txt', '1000074.txt', '1000250.txt', '1000289.txt', '100066.txt', '1000709.txt', '1000790.txt', '1001106.txt', '1001668.t
Total documents: 9943
可用詞彙數量: 3733

前 20 個詞:
1. aaa
2. ab
3. ac
4. ae
5. aid
6. alan
7. alt
8. an
9. and
10. asia
11. au
12. auto
13. bank
14. bc
15. beta
16. body
17. bold
18. book
19. bot
20. box

後 20 個詞:
3714. 高級
3715. 高速
3716. 高速公路
3717. 高速鐵路
3718. 高達
3719. 高鐵
3720. 高雄
3721. 鬥爭
3722. 鳳凰
3723. 鴉片
3724. 麥克
3725. 麻省理工
3726. 麻薩
3727. 黃金
3728. 黎曼
3729. 黑色
```