以民事訴訟之爭點分群為基礎的類似案件搜尋系統
Clustering Issues in Civil Judgments for Recommending Similar Cases

劉一凡　　劉昭麟　　楊婕
Yi-Fan Liu　　Chao-Lin Liu　　Chieh Yang

國立政治大學資訊科學系
Department of Computer Science, National Chengchi University
{108753213, chaolin}@g.nccu.edu.tw, 05141343@gm.scu.edu.tw

摘要

類似案件搜尋是法律實務中十分重要的任務，從中能獲取珍貴的法律見解。而爭點是民事訴訟中兩造互為對立的主張，代表審理案件時要考慮的核心事項。許多研究以不同角度計算判決書間相似度；而我們提出以爭點的分群編碼判決書的方法，來建構一個類似案件搜尋系統。我們以具有法律背景的人工評分來驗證系統的有效性，同時比較數種前處理程序和分群方法不同組合所達成的效果。

Abstract

Similar judgments search is an important task in legal practice, from which valuable legal insights can be obtained. Issues are disputes between both parties in civil litigation, which represents the core topics to be considered in the trials. Many studies calculate the similarity between judgments from different perspectives and methods. We first cluster the issues in the judgments, and then encode the judgments with vectors for whether or not the judgments contain issues in the corresponding clusters. The similarity between the judgments are evaluated based on the encoded messages. We verify the effectiveness of the system with a human scoring process by a legal background assistant, while comparing the effects of several combinations of preprocessing steps and selections of clustering strategies.

關鍵字：分群、資訊檢索、語意搜尋、法資訊學、類似案件

Keywords: clustering, information retrieval, semantic search, legal informatics, similar legal cases

1　緒論

判決書中富含法院對於法律問題的見解，而所謂「類似案件」是其中兩者在某個面向上類似且能提供有用資訊的判決書。法官在撰寫判決書、訴訟當事人及律師在準備攻防時，需要查找其他類似案件作為參考；法律實證研究者則需要蒐集類似案件以研究法律對於社會的影響及執行成效；民眾則可透過閱讀類似案件了解某些司法實務。如何有效地查找類似案件是基礎且重要的工作。

爭點，是兩造矛盾或互為對立的主張，包含事實上及法律上的爭議事項，一個案件中常含有數項不同爭點；法院會對這些爭點做出判斷，最終作出判決。爭點在法院的準備程序被統整並記載於筆錄，並不會公開；然而法官書寫判決書時經常將其作為論述的核心，記載於判決書中。爭點作為案件的審查要點，我們認為十分適合作為一種類似案件的面向。

一般的判決書檢索系統只能以使用者提供的關鍵字在全文或段落中搜尋。若使用者對一篇案件產生興趣，想要查看其他類似的案件，僅能依照自己的既有知識重新提供關鍵字搜尋，既不方便更無法獲得來自系統的額外資訊。

在這份研究中，我們提出以不同案件中爭點的分群 (clustering) 來編碼判決書，並以編碼後的匹配程度作為案件相似度的類似案件搜尋系統，其功能可以為一般判決書檢索系統推薦類似案例，希望能解決上述困境。

我們選擇三個系統元件當作研究變項，實驗並探討兩種資料萃取、三種文本向量化 (text vectorization)、兩種分群方法對於系統的影響。評估方面，一位法律系畢業的專任研

究助理為系統提供的數據做三種等級的標記，我們以此比較不同方法實作系統元件的效果。

2 相關研究

法律結合科技一向是熱門研究主題，近期國內法界學者開始進行人工智慧引入民事程序的可行性研究 (Ho, 2021)，越來越多機會正在產生。而法律資訊檢索的研究對象包含各式法律文獻間的搜尋，始終圍繞相似性這個主題。而 Bhattacharya et al. (2019) 對於研究法律案件相似性的方法提出總結，方法主要一是以引用為基礎 (citation-based) 計算；二是以文本為基礎 (text-based) 計算，包含使用全部文件、段落連結 (paragraph links)、主要論題相似性 (thematic similarity)、摘要等方法。

我們能以這樣的架構回顧更早期的研究。Kumar et al. (2011) 的研究表明使用法律詞彙相較於全部詞彙計算相似度對於尋找類似案件有更好效果。另外法律文書的書目耦合能加強共被引方法的效果；Raghav et al. (2016) 利用分群技術找出段落連結，結合案件引用的資訊計算案件的相似度。

然而，相對於採取判例法 (case law) 的海洋法系國家，我國法院裁判書並沒有這些引用判例的資訊。因此我們可以著重在如何更好地提取出文本的法律特徵。Ma et al. (2018) 利用法律知識圖譜將中文判決書提取為法律概念並學習相似度； Hong et al. (2020) 對中國判決書結構和類似案件匹配任務的挑戰進行分析，並提出結合法律特徵向量及預訓練語言模型。

許多學者對國內法資訊學的發展做出貢獻，這些不同任務的研究往往也聚焦在更好地提取及利用法律文本資訊，也可供我們效法。其中 Lin et al. (2012) 嘗試自動擷取 21 種針對強盜罪與恐嚇取財罪定義之標籤並利用於案件分類和量刑預測； Liu and Chen (2019) 提出能自動萃取出裁判書要旨句的模型，實驗多種類神經網路模型架構及特徵選擇的效果。除此之外，歷年來國內許多實作裁判書檢索系統的碩博士學位論文也能提供借鏡。Lan (2009) 提出將關鍵詞檢索結果以階層式分群法輸出，及共現詞彙建立索引的檢索系統；Lu (2021) 提出以空間向量模型合併 TF-IDF 詞權重調整之檢索系統；Tsao (2021) 以預訓練語言模型建立判決書的情境表示式，並提出案由分群亂度當作實驗指標。

3 問題定義與假設

$\boldsymbol{D}$ 表示包含 D 的所有判決書的集合。我們的目標是建立一個系統 f，能夠找出 D 的一些類似案件 $\boldsymbol{S}$。

$$D \in \boldsymbol{D} \tag{1}$$

$$f(D) \rightarrow \boldsymbol{S} \tag{2}$$

我們假定判決書 D 中的爭點具有足以代表該判決書的核心資訊，且具有越多相似爭點的判決書則越相似。因此，我們設計以下流程和定義：判決書首先以萃取出爭點的方法 e 取得爭點列表，隨後對其進行前處理 p；接著使用文本向量化的方法 v 將其轉換為數字向量；之後，我們使用分群方法 c 對所有判決書中的爭點進行分群；最後把每一個群 (clusters)以自然數編號後，將每一篇判決書中的爭點代換為其所屬之群的號碼，此過程稱為代換 r；這一組數字稱為群代碼 C。

至此，我們能將原始判決書的文本 D，經過一系列方法轉換為群代碼 C。我們將經過 $e, \mathrm{p}, v, c, \mathrm{r}$ 的過程合稱為分群編碼 t。

$$t: \quad D \xrightarrow[e,\mathrm{p},v,c,\mathrm{r}]{} C \tag{3}$$

類似案件定義為：C_1, C_2 為 S_1, S_2 的群代碼，若且為若 C_1, C_2 的交集數大於等於閾值 θ，則 S_1 與 S_2 互為類似案件。一群互為類似案件的集合則表示為 $\boldsymbol{S}$。

$$C_1 = t(S_1),\ C_2 = t(S_2) \tag{4}$$

$$S_1 \sim S_2 \ \Leftrightarrow\ |C_1 \cap C_2| \geq \theta \tag{5}$$

$$\boldsymbol{S} = \{S_1, S_2, \cdots\} \tag{6}$$

基於上述，我們將此類似案件的搜尋系統以分群編碼 t 實踐，記為 f_t。

我們想知道，具法律背景人士如何評價系統所找出的類似案件，並測試、比較系統中的一些不同方法產生之效果，以找出未來改進系統的方向。為此，我們以 2 種 e、3 種 v、2 種 c 組合成的 $t_1, t_2, \cdots, t_{12}$（搭配固定的 p、r），構建出總共 12 個系統 $f_{t_1}, f_{t_2}, \cdots, f_{t_{12}}$。評估方法於 9.2 節說明。

4 資料來源與篩選

司法院資料開放平臺[1]提供民國 85 年起至今超過千萬筆的判決書，每月持續更新。我們下載並篩選出案號字別[2]為「勞訴」，代表第一審勞動訴訟事件（以下簡稱勞訴）的判決書，時間分布自民國 88 年至 110 年為止，共 15267 篇。這些資料並沒有註明彼此的關係，每一篇僅提供法院、年分、日期、案號字別、案由及判決書文本；而民事訴訟法第 226 條[3]僅規定判決書必須出現的一些事項，其餘事項與格式則由法官依習慣及自由決定。因此判決書寫作上雖然存在一些常見的規律，但並沒有普遍適用的格式，可以視為富含資訊的非結構化的文本。為了簡化研究，我們不會使用全部的勞訴，而是以下面兩個步驟進一步篩選出具有共同性的研究資料。

4.1 步驟一：爭點段落

目前並沒有能普遍適用的方法可以定位出判決爭點，所以我們先聚焦在找出含有「爭點段落」的勞訴，定義及流程如下：首先以資料集內固定的數種章節編號（一、, 甲 …等）和出現於行首的條件分段，將分段所得的語料稱為「章節分行」；進一步觀察發現，爭點段落的標題常具有固定模式，通常會包含「爭點」、「爭執(之)(事項|要旨|重點)」屬於正面的關鍵詞，不包含「不爭執」、「其餘」屬於反面的關鍵詞；爭點段落的下一段則會回到和開頭同一種章節編號的下一個編號，可以此定位爭點段落的結尾。我們以正規表示法[4] (regular expressions) 比對這些模式，定位出爭點段落的開頭與結尾，找出明確含有爭點段落的判決書共 5060 篇。然而，爭點段落的內容依不同法官風格而定，並不是含有爭點段落的判決書都適合拿來利用；我們將搭配下一個步驟進一步篩選語料。

反問	發問
含有關鍵詞：為什麼|還要|怎可能|孰能|遑論|難道|如何能|又如何|何需|何須|何必|豈|否則|明知|何來|試問|焉|何以。	含有關鍵詞：問|說|證稱|你|我|們|嗎|所以|對不對|啊|…|那是|然後|那。 搭配前後的引號。

表 1. 反問及發問的模式

4.2 步驟二：爭點問句

不同判決書的爭點段落仍有各式不同寫法，可能包含當事人主張、法院見解、不同格式的爭點。為了找出具有普遍性、能提供充分語意的語料，我們選擇篩選出在爭點段落內具有符合「爭點問句」定義的判決書作為語料，共 3837 篇。

所謂爭點問句，定義為：以句號做為結尾且以正規表示法，排除模式上明顯為寫作上的反問語氣及言詞辯論程序記錄的發問。表 1 紀錄上述兩者的模式。設計爭點問句的後半部定義，其目的為雜訊抑制 (noise reduction)，即使不能完善也對提升語料品質有所助益。且上述須排除的反問及發問並不常見於爭點段落內；因此，即使不能保證全部排除，仍可確保語料擁有較高的品質。

5 資料萃取

原始判決書以 4.1 節所述分段方法切割為數個章節分行後，進一步觀察章節分行內的結構，能發現法院在列舉爭點時，時常將相關的爭點問句以群組的形式記錄在同一章節分行，例如：「一、被告對原告為解雇處分之事由為何？該解雇處分是否適法？有無逾越勞基法及被告聘雇人員工作規則所定除斥期間？」、「二、原告得否向被告請求退休金？得請求之金額為若干？」；有時也會發生爭點問句與論述夾雜的情況，若不對章節分行切割則無

[1] https://opendata.judicial.gov.tw/

[2] https://law.judicial.gov.tw/GetFile.ashx?pfid=0000309583

[3] 民事訴訟法第 226 條：「1 判決，應作判決書，記載下列各款事項：一、當事人姓名及住所或居所；當事人為法人、其他團體或機關者，其名稱及公務所、事務所或營業所。二、有法定代理人、訴訟代理人者，其姓名、住所或居所。三、訴訟事件；判決經言詞辯論者，其言詞辯論終結日期。四、主文。五、事實。六、理由。七、年、月、日。八、法院。2 事實項下，應記載言詞辯論時當事人之聲明，並表明其聲明為正當之攻擊或防禦方法要領。3 理由項下，應記載關於攻擊或防禦方法之意見及法律上之意見。4 一造辯論判決及基於當事人就事實之全部自認所為之判決，其事實及理由得簡略記載之。」。

[4] https://pypi.org/project/regex/

法萃取出資料。這讓我們必須考量在「拆開個別子句」和「保持原子句群組」兩種做法間，有取得更單純語意和保持語料相關性及完整性之取捨；因此我們設計兩種萃取爭點問句的方法，在具有爭點問句的判決書 3837 篇中，以方法一 NS 處理後每篇平均有 3.3 句爭點問句，每句平均有 46.8 字；以方法二 EX 處理後每篇平均有 4.8 句爭點問句，每句平均有 32.7 字。

5.1 方法一：NS

第一型保留法院原始的爭點問句群組，只篩選該章節分行是否為爭點問句，定義於 4.2 節。其優點如上所述能保持相關性及完整性；缺點則為擁有更複雜的語意，且少數情況下可能夾帶法院見解或是當事人主張。我們將此種資料萃取方式稱為 NS，代表 Non-Split 之意思。

5.2 方法二：EX

第二型考量為有利取得更單純語意之句向量，以及盡可能找出被包覆在其他無關資訊中的爭點問句，先將章節分行做「分句」處理，再進行爭點問句篩選。分句主要以章節編號和三種具有置於完整語意句末的標點「。！？」來切割章節分行。此種方法會將一個原先群組化的爭點問句拆分成數個子句，缺點是可能分句出語意太狹隘子句，例如「有無理由？」、「金額若干？」。我們將此種資料萃取方式為 EX，代表 Extracted 之意思。

5.3 抽樣及分析

以上兩種萃取爭點問句的方法各有其優劣處，為了進一步了解它們所帶來的誤差，我們簡單隨機抽樣 3837 篇中的 383 篇，以人工檢驗試圖了解萃取方法的效果。抽樣的結果中 NS 方法有 56 篇 (16.4%)、EX 方法有 27 篇 (7%) 有未能完全萃取出所有爭點問句或萃取錯誤的情形，其誤差範圍大多數在二句之內。

進一步分析這些錯誤樣本，發現造成的理由大致可分為四類：1. 寫作風格（法官將理由與爭點書寫於同一段而重複提及該爭點問句導致冗餘；部分爭點不以問句的方式呈現；爭點問句被夾在長句子當中等）、2. 分段誤差（涵蓋到額外的段落）、3. 錯誤切割子句（同一個句子中使用到章節編號時被意外切割為二句）、4. 錯誤排除（反問語氣及言詞辯論程序記錄的發問）。四類理由中法官寫作風格差異為最大宗，分別佔 83.9% (NS) 與 66.6% (EX)。

原始	變換後
㈢教師法第十四條第一項第六款行為不檢有損師道…	教師法第 14 條第 1 項第 6 款行為不檢有損師道…
㈠原告係 79 年 7 月 13 日或 84 年 4 月 28 日起受被告僱用？	原告係自某時或某時起受被告僱用？
(一)世新視訊股份有限公司與被告公司是否具有實體同一性？	某團體與被告公司是否具有實體同一性？
（一）王淑芬之死亡是否屬職業災害？	某人之死亡是否屬職業災害？

表 2. 原始及變換後之爭點問句

6 資料前處理

6.1 法規條文正規化

爭點問句中記載的法規條文，常因不同的書寫習慣或簡稱而導致缺乏一致的形式，例如「勞資爭議處理法第一條」和「勞資法第 1 條」具有相同意義。若是不對其正規化會導致機器將其視為不同單元而造成錯誤分群。

為此，我們將其中文數字代換為阿拉伯數字，以及利用從全國法規資料庫[5]蒐集的法規條文和自行建立的字典將簡稱代換為原本的名稱。

6.2 細節模糊化

排除掉上述法規條文，爭點問句常常有過多私人性質的訊息，如人名、地址、數字細節等。為了使其更具一般性，我們希望將這些訊息模糊化，意即將它們轉為文字代號。

首先是關於數字細節。章節編號、金額、時間、算式等能以正規表示法找出的數字細節被替換成統一的文字代號，如某金額、某時等。再來是運用命名實體辨識 (named-entity recognition) 技術替換細節的方法。除了數字細節，人名、地址、某團體等不易以規則辨識的命名實體也需要模糊化。我們選用中研院開源的 CKIP Transformers[6]套件之 bert-base-chinese-ner 模型進行命名實體辨識，將辨識所得替換成某人、某地、某團體等文字代號。經過上述程序變換後的一些例子如表 2。

[5] https://law.moj.gov.tw/

[6] https://github.com/ckiplab/ckip-transformers

6.3 斷詞

詞彙是文本中語意的最小單位。詞袋模型 (bag-of-words model) 將文本表達為一群詞彙的組合，不考慮文法及順序，並可以詞彙出現的頻率或其他方式做為特徵。然而中文在書寫上不像英文以空白區分出詞彙；為了使用詞袋模型來表達文本，必須先將其斷詞 (word segmentation)。我們使用 CKIP Transformers[7]套件中的 bert-base-chinese-ws 模型，並搭配上述兩個小節調整斷詞結果，使得法規條文和文字代號不被斷開。

7 文本向量化

為了讓機器能識別文本中所蘊含的文意，需要將文本轉化為數字形式，且能表現出該文本語意上、語言學上的特徵，稱為文本向量化 (text vectorization)。我們選用兩種常見的文本向量化方式來轉化語料以便進行分群。一是基於深度學習的句嵌入 (sentence embedding) 技術；二是基於詞袋模型擴展出的 n-gram 模型搭配資訊檢索領域中常用的 tf-idf 加權技術 (term frequency-inverse document frequency weighting)。這些技術在接下來的小節提供介紹。

7.1 Sentence-BERT(SBERT)

Reimers et al. (2020) 以孿生類神經網路（Siamese neural network）及三連體類神經網路（Triplet neural network）架構修改預訓練 BERT 語言模型 (Devlin, 2018)，以得到具有語意、能以 cosine similarity[8] 比較相似度的的句嵌入技術。

孿生類神經網路是以兩個共享權值 (weights) 的子類神經網路所建構而成 (Bromley, 1993)。將兩筆資料輸入進兩個類神經網路進行特徵轉換 (feature transformation)、特徵提取 (feature extraction) 後，以 loss function 計算兩者的相似度。而三連體類神經網路則是以前者改進，改為三個輸入與子類神經網路，藉由正樣本與負樣本的組合來計算。

SBERT 將標記好相似性的句子對做為訓練資料，將 transformers 類的網路結合上述類神經網路，比較多種目標函數及整合特徵向量的效果，產生出一批預訓練語言模型，能夠將一定長度內的句子輸出成固定維度的句向量。他們發現與 BERT 相比，在一大群句子（約一萬筆）中尋找最相似句子對的任務上能大幅度節省時間（65 小時縮減至 5 秒鐘），並且在諸多語意相似度 (semantic textual similarity) 任務上成為最先進的 (state of art) 模型。由於上述特性，SBERT 更適合我們的分群任務上。實作上，使用 sentence-transformers[9]套件，並選用較為貼近實驗設計的 paraphrase-multilingual-mpnet-base-v2 模型。由於缺乏成對的相似句組能作為訓練資料，我們沒有進行 fine-tune。

7.2 TFIDF-RAW

tf-idf 由統計得來的詞頻（term frequency, tf）及逆向文件頻率（inverse document frequency, idf）組成，兩者相乘可以表示詞彙的重要程度。一般來說，某詞彙在特定文件中的詞頻計算方式為在該文件中該詞彙出現的次數，除以該文件中所有詞彙出現次數之合；某詞彙的逆向文件頻率計算方式則為總文件數目除以包含該詞彙之文件的數目，再取以 10 為底的對數 (Jones, 1972)。

將句子以詞袋模型表示，搭配 tf-idf 加權技術，可得到固定維度的句向量。我們以 TF-IDF-RAW 表示這種向量化方法。

實作方面，我們使用 scikit-learn 套件中的 TfidfVectorizer[10]。設定其所提供的一些參數：norm='l2', use_idf=True, smooth_idf=True, sublinear_tf=False，這會讓 idf 的計算方式成為：

$$\text{idf(t)} = \log\frac{\text{n}+1}{\text{df(t)}+1} + 1 \qquad (7)$$

n 代表總文件數，df(t) 代表包含詞彙 t 的文件數。為了避免除數為零錯誤 (zero division error) 及平滑化 (smoothing)，分子及分母都加上數字 1。最後會對計算出的 tf-idf 向量做 l2 正規化 (normalization)：

$$v_{norm} = \frac{v}{\sqrt{v_1^2+v_2^2+\cdots+v_n^2}} \qquad (8)$$

[7] https://github.com/ckiplab/ckip-transformers

[8] cosine similarity: $\text{c(a, b)} = \frac{a\cdot b}{|a||b|}$

[9] https://www.sbert.net/

[10] https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html

另外值得一提的是，設定中 analyzer='word'，使得輸入的字串根據空白切割出詞彙（為此要準備詞彙間以空白隔開的句子）。設定 token_pattern=r"(?u)\b\w+\b" 使其能以空白分割出詞彙且不會過濾掉長度為 1 的中文詞彙。我們沒有使用停用詞 (stop words)。

7.3 TFIDF-NGRAM

n-gram 模型可以視為詞袋模型的擴展，它將 n 個連續詞彙組合成新的詞彙，以此捕捉連續詞彙之間的關係，再將文本表達為一堆詞彙的集合。我們以 TFIDF-NGRAM 表示使用 tf-idf 技術同時搭配 unigram+bigram+trigram (分別代表 n=1,2,3) 為特徵的向量化方法。實作工具及設定與上一段基本相同，額外設定 ngram_range=(1,3) 以使用 n-gram 模型。

8 分群與搜尋機制

我們設計的類似案件搜尋系統需要利用分群結果，並以其對文本（爭點問句）進行「分群編碼」。這一節將會先介紹我們所使用的兩種分群演算法，再說明使用分群結果得到群代碼及以群代碼搜尋出類似案件。

8.1 Affinity Propagation (AP)

AP 是基於資料點間相互資訊傳遞求得群集中心，再以此得出不同群的分群演算法 (FREY, 2007)。步驟是初始化各資料點間的責任值（Responsibility）、可用值（Availability），計算任兩資料點間的上述數值，重複迭代直到各資料點收斂。s(i,k)代表 i,k 兩點的相似度，則對資料點 i, k 來說，責任值 r 的更新公式為：

$$r(i,k) \leftarrow s(i,k) - \max[a(i,k') + s(i,k') \forall k' \neq k] \quad (9)$$

可用值 a 的更新公式為：

$$a(i,k) \leftarrow \min[0, r(k,k) + \sum_{i' s.t. i' \notin \{i,k\}} r(i',k)] \quad (10)$$

透過給定參考度（Preference）而不指定分成幾群，得到基於資料特性的分群結果，我們將其設定為所有輸入向量彼此相似度之平均。為了優化結果，我們進行 grid search 後決定以下的 hyper-parameters: convergence_iter=15, max_iter=1000, damping=0.9, random_state=0。affintiy 則根據向量化方法，TFIDF-RAW, TFIDF-NGRAM 為"euclidean [11] "，SBERT 為 "precomputed"，事先計算其 Cosine Similarity。

8.2 Hierarchical Clustering (HC)

HC 是一種分群演算法的類型，以資料點之間的距離計算出不同的群。主要分成 1. 首先所有資料視為一群，再依距離一一分出不同群。2. 把每筆資料視為一群，自下而上聚合，得到樹狀結構的分群結果。若不設定停止條件，最終根節點是聚合所有樣本的單一群，所有葉節點則是只含有一個樣本的群。我們使用自下而上的聚合方法 (agglomerative)。為了優化結果，以 grid search 決定 hyper-parameters：TFIDF_RAW, TFIDF_NGRAM 二者同被設定為 affinity='euclidean',linkage='ward',distance_threshold=1.9; SBERT 則被設定為 affinity='cosine', linkage='complete', distance_threshold=0.3。

8.3 分群編碼與類似案件搜尋

當我們以上述兩種方法得到分群結果後，把每一個群以自然數編號，將判決書中的爭點問句代換為其所屬的群代碼。至此，我們完成從原始判決書到群代碼的轉換，而這一系列的流程稱為分群編碼。

我們可以選定其中一篇為查詢 (query)，計算它與其他文件的群編碼交集數，以交集數由大到小排序，作為返回的搜尋結果，這稱為類似案件搜尋。

9 實驗設計

9.1 研究限制

由於我們研究的是真實世界資料以及嶄新的類似案件定義，並不事先存在這樣的標記可以使用，且基於人力、經費等限制，我們無法事先準備類似案件、分群的標準答案以進行監督式學習。因此，實驗設計以標記者對系統提供的答案進行評估，得到兩兩案件間相似度的標記資料後，衡量不同實驗變項之系統的成效差異。在未來，我們能以這些標記資料訓練模型以改善類似案件搜尋系統。

[11] Euclidean distance: $d(p,q) = \sqrt{(q_1 - p_1)^2 + \cdots + (q_n - p_n)^2}$

9.2 實驗變項

我們共選擇三個實驗變項，其中欲比較的方法說明理由如下：

資料萃取方面，比較第 5 節所述的兩種方法：NS 與 EX。兩者的關鍵差別在於爭點問句群組化與否，導致語意複雜程度、語料相關性及完整性的差距。我們想知道對系統而言的影響程度及選擇何者。

文本向量化方面，比較第 7 節所述的三種方法：SBERT, TFIDF_RAW, TFIDF_NGRAM。我們想知道近期的句嵌入技術與傳統詞袋模型及統計為基礎的方法，在系統中的效果以及改善方法為何。

分群演算法方面，使用第 8 節所述的兩種演算法：AP 與 HC。我們希望盡可能地以資料特性決定分群結果，因此這兩種選用的分群演算法都不是直接決定群的數量，而是藉由設定能決定資料點間是否為一群的標準。這個標準在 AP 裡是參考度，設定為所有資料點的相似度平均 (SBERT 以 cosine similarity 計算，TFIDF-RAW, TFIDF-NGRAM 則以 Euclidean distance 計算)；在 HC 裡則是相似度 (SBERT 以 cosine similarity=0.3，TFIDF-RAW, TFIDF-NGRAM 則以 Euclidean distance=1.9 計算)。由於語料經過不同實驗變項組合的處理，以上述的標準分群後會產生出不同的分群結果。我們將這些分群結果之群的數量紀錄在表 3。

9.3 評估方法

將上述實驗變項組合，得到 12 種不同分群編碼的系統。以相同流程評估這些系統，如下：

第一步，將資料集內每一篇判決書作為查詢進行類似案件搜尋，會得到 3837 個（資料集大小）搜尋結果。第二步，提取出每組最高分的搜尋結果，經過「篩選」後得到數組「提問與推薦」的組合，稱為推薦案件組，接著經過資料庫的比對，以避免重複評分。篩選的辦法希望排除 1. 個別法院內對於爭點整理的固有習慣。2. 類似程度不高的判決書。因此設計過濾規則：1.來自相同法院 2. 分群交集數小於 3 的推薦案件組。這裡的 3 即為第 3 節中的閥值 θ。第三步，請一位法學系畢業的專任助理（簡稱為標記者），以其中各自爭點問句的類似程度，對推薦案件組作三種等級的評分，分別是：比較類似、勉強類似、不類似，隨後將標記過的推薦案件組評分記錄到資料庫中。以上三項步驟產生出不同方法組合的評分統計結果於表 4。

方法組合	群數
EX + AP + SBERT	883
EX + AP + TFIDF-NGRAM	2421
EX + AP + TFIDF-RAW	1603
EX + HC + SBERT	1390
EX + HC + TFIDF-NGRAM	1446
EX + HC + TFIDF-RAW	1374
NS + HC + SBERT	807
NS + HC + TFIDF-NGRAM	813
NS + HC + TFIDF-RAW	877
NS + AP + SBERT	630
NS + AP + TFIDF-NGRAM	1559
NS + AP + TFIDF-RAW	1094

表 3. 不同實驗變項之群的數量

接下來我們可以就評分統計中不同評分所占的比例來對不同面向做討論，做為評估方法。兩個方法組合相比，有較高比例的比較類似與較低比例的不類似者，我們定義其為較佳的表現，反之則為較差的表現。

10 實驗數據與討論

首先觀察表 3 群數和表 4 不同方法合計的案件數的關係：可以看出若群數越少則合計的案件數越多。這是由於在篩選推薦案件組的過程會根據 8.3 節所計算的交集數，而較少的分群數則更容易產生群代碼的交集。

接下來，我們將分別以資料萃取、向量化、分群方法的三個面向來比較其在系統中的表現，以及探討所造成的原因。

首先，比較不同資料萃取方法：大致上 NS 比起 EX 有著稍微較佳的表現。我們認為這顯示 NS 保留法院所提供群組化的爭點問句，其句向量的語意更為豐富，因此分群的結果更為細緻，從而後續步驟所得的推薦案件組較能得到標記者的青睞。

再來，比較三種向量化方法： TFIDF-RAW 比起 TFIDF-NGRAM 皆得到較高比例的比較類似、較低比例的不類似。我們認為這是由於 TFIDF-NGRAM 雖然更能考慮詞彙的相鄰性，但其較大的維度反而不利相似度計算；而 SBERT 和另外兩向量化方法相比，能提供較多推薦案件組及更穩定的表現。我們認為這要歸功於其將不同詞彙但意思相近之句子

評分等級 / 方法組合	比較類似		勉強類似		不類似		所有標記
	數量	比例	數量	比例	數量	比例	數量
EX + AP + SBERT	181	55.7%	70	21.5%	74	22.8%	325
EX + AP + TFIDF-NGRAM	51	56.0%	15	16.5%	25	27.5%	91
EX + AP + TFIDF-RAW	103	59.5%	41	23.7%	29	16.8%	173
EX + HC + SBERT	449	54.9%	176	21.5%	193	23.6%	818
EX + HC + TFIDF-NGRAM	178	36.4%	77	15.7%	234	47.9%	489
EX + HC + TFIDF-RAW	121	52.6%	44	19.1%	65	28.3%	230
NS + HC + SBERT	204	55.4%	82	22.3%	82	22.3%	368
NS + HC + TFIDF-NGRAM	75	27.0%	28	10.1%	175	**62.9%**	278
NS + HC + TFIDF-RAW	45	63.4%	10	14.1%	16	22.5%	71
NS + AP + SBERT	60	60.6%	25	25.3%	14	14.1%	99
NS + AP + TFIDF-NGRAM	18	60.0%	4	13.3%	8	26.7%	30
NS + AP + TFIDF-RAW	38	**67.9%**	11	19.6%	7	12.5%	56
平均	126.9	50.3%	48.6	19.3%	76.8	30.4%	252.3

表 4. 評分統計表

嵌入為相似向量的能力，以及它較小的特徵維度。

最後，關於兩種分群演算法：我們注意到 AP 相較於 HC 得到較好且較穩定的結果；然而不能排除這是受到分群演算法相關參數的影響所導致，因此不能以此宣稱 AP 是適合本任務的分群演算法；我們仍可以得到不同分群演算法及其設定對於類似案件搜尋結果有較大影響的結論。

值得注意的是，EX + HC + TFIDF-NGRAM 與 NS + HC +TFIDF-NGRAM 分別得到使用 EX 和 NS 的所有方法組合中最差的表現，且與其他方法組合的表現差距甚大。我們認為造成此現象的原因為， TIDF-NGRAM 有較高的為度，而 HC 在高維度時計算 Euclidean distance 所得的相似性效果較差，SBERT 則計算 cosine similarity 而受到影響較少。

整體而言，12 個不同變項組合的系統所取得的類似案件組平均有 50.3% 的比較類似、30.4% 的不類似；而我們實驗組中的最好結果則有 67.9% 的比較類似、12.5% 的不類似。考量判決書中爭點本身的多樣性、相似性及重複率尚屬未知，我們認為這樣的類似案件系統具備有效性，且能夠提供往後研究者一個基準以及研究方向。

11 結論

在這份研究中，我們設計了一套以民事訴訟之爭點分群為基礎的類似案件搜尋系統，並且嘗試比較 12 組不同資料萃取、向量化、分群方法對系統的影響。以具有法律背景的人工評分結果顯示，我們表現最好的系統，所找到的類似案例組中 67.9% 被評為比較類似(最高等級的評分)，僅有 12.5% 被評為不相似；而 12 組系統平均有 50.3% 的被評為比較類似、30.4% 的不類似，顯示我們的方法具備一定的有效性，足以作為後續研究的基準。而這些實驗所得的標記資料，能開啟未來監督式學習的研究路徑。

致謝

本研究承國科會研究計畫 107-2221-E-004-009-MY3 與 110-2221-E-004-008-MY3 與國立政治大學高教深耕校內補助計畫 111H124D-13 之部分補助，謹此致謝。

References

Paheli Bhattacharya, Kripabandhu Ghosh, Arindam Pal, Saptarshi Ghosh. 2019. Methods for Computing Legal Document Similarity: A Comparative Study. *LDA 2019 workshop.* the Fundation for Legal Knowledge Based System. https://doi.org/10.48550/arXiv.2004.12307

Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Sickinger and Roopak Shah. 1993. Signature Verification using a "Siamese" Time Delay Neural Network. *Advances in Neural Information Processing Systems 6*, pages 737-744. Neural Information Processing Systems foundation https://doi.org/10.1142/S0218001493000339

Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Computing Research Repository,* arXiv.1810.04805

Brendan J. Frey , Delbert Dueck. 2007. Clustering by Passing Messages Between Data Points. *SCIENCE, 315*(5814)972-976. https://doi.org/10.1126/science.1136800

Jim-How Ho. 2021. AI 引入民事程序可行性之研究 (The Feasibility Research on Introducing Artificial Intelligence into Civil Procedures) [In Chinese] Doctoral Dissertation, Department of Information Management, National Taiwan University of Science and Technology. https:/hdl.handle.net/11296/pkvh27

Zhilong Hong, Qifei Zhou, Rong Zhang, Weiping Li, Tong Mo. 2020. Legal Feature Enhanced Semantic Matching Network for Similar Case Matching. *2020 International Joint Conference on Neural Networks* pages 1-8. the Institute of Electrical and Electronics Engineers.https://doi.org/10.1109/IJCNN48605.2020.9207528

Karen Spark Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation, 28*(1), pages 11-21. https://doi.org/10.1108/eb026526

Sushanta Kumar, P. Krishna Reddy, V. Balakista Reddy, Aditya Singh. 2011. Similarity analysis of legal judgments. *COMPUTE '11: Proceedings of the Fourth Annual ACM Bangalore Conference* pages 1-4. Association for Computing Machinery. https:doi.org/10.1145/1980422.1980439

Chia-Lian Lan. 2009. 中文訴訟文書檢索系統雛形實作 (A Prototype of Information Services for Chinese Judicial Documents)[In Chinese]. Master's Thesis, Department of Computer Science, National Chengchi University. https://hdl.handle.net/11296/hgfrwt

Chao-Lin Liu, Kuan-Chun Chen. 2019. Extracting the Gist of Chinese Judgments of the Supreme Court. *ICAIL '19: Proceedings of the Seventeenth International Conference on Artificial Intelligence and Law* pages 73–82. Association for Computing Machinery. https://doi.org/10.1145/3322640.3326715

Wan-Chen Lin, Tsung-Ting Kuo, Tung-Jia Chang, Chueh-An Yen, Chao-Ju Chen, and Shou-de Lin. 2012. 利用機器學習於中文法律文件之標記、案件分類及量刑預測 (Exploiting Machine Learning Models for Chinese Legal Documents Labeling, Case Classification, and Sentencing Prediction) [In Chinese]. *December 2012-Special Issue on Selected Papers from ROCLING XXIV. 17.* International Journal of Computational Linguistics & Chinese Language Processing. https://aclanthology.org/O12-5004

Kai-Yu Lu. 2021. 基於向量空間模型之智慧型文件搜尋系統開發–以台灣醫療糾紛判決書為例 (Development an Intelligent Document Search System Based on Vector Space Model - A Case Study of Taiwan Medical Malpractice Claim Judgment) [In Chinese]. Master's Thesis, Department of Medical Informatics, Chung Shan Medical University. https://hdl.handle.net/11296/ceu267

Yinglong Ma, Peng Zhang, Jiangang Ma. 2018. An Efficient Approach to Learning Chinese Judgment Document Similarity Based on Knowledge Summarization. *Computing Research Repository,*. arXiv.1808.01843

K. Raghav, Pailla Balakrishna Reddy, V. Balakista Reddy, Polepalli Krishna Reddy. 2016. Text and Citations Based Cluster Analysis of Legal Judgments. *Mining Intelligence and Knowledge Exploration. 9468*, pages 449–459. International Conference on Mining Intelligence and Knowledge Exploration. https://doi.org/10.1007/978-3-319-26832-3_42

Nils Reimers and Iryna Gurevych. 2020. Making Monolingual Sentence Embeddings Multilingual using Knowledge Distillation. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing.* Association for Computational Linguistics. https://arxiv.org/abs/2004.09813

Hsi-Chang Tsao. 2021. 基於深度學習模型之判決書情境相似檢索技術之研究(Research on Similar Situation Retrieval Technique for Court's Judgment Based on Deep Learning Model) [In Chinese]. Master's Thesis, Department of Computer Science and Engineering, National Chung Hsing University. https://hdl.handle.net/11296/gjs6z4