國立臺灣師範大學 資訊工程研究所

指導教授：陳柏琳博士

使用機器學習方法於語音文件檢索之研究

Exploiting Machine Learning Methods for Spoken Document Retrieval

研究生：游斯涵　撰

中 華 民 國　　九十八　　年　　二 月

摘要

本論文初步地討論機器學習之方法在資訊檢索上的應用，即所謂排序學習(Learning to Rank)；並針對近年被使用在資訊檢索上的各種機器學習模型及概念，以及所使用的各種特徵，包含詞彙本身之特徵、相近度特徵、及機率特徵等進行分析與實驗。除此之外，本論文亦將之延伸至語音文件檢索的應用上。本論文初步地使用TDT(Topic Detection and Tracking)中文語料部份作為實驗題材，此語料為過去TREC(文件檢索暨評測會議)上公開評估語音文件檢索系統的標準語料(Benchmark)之一，此語料包含TDT-2及TDT-3兩套語料，提供了大量的新聞語料，及豐富的主題、轉寫等標註，以作為語音文件檢索相關研究使用。為了更有效地開發富含資訊的語音文件特徵，本論文亦使用臺師大大陸口音中文大詞彙連續語音辨識器(Large Vocabulary Speech Recognition, LVCSR)作為語音文件轉寫平台，產生的詞圖(Word Graph)，作為擷取語音文件獨特特徵的主要依據。此外，我們並考慮到資訊檢索中之訓練語料不平衡問題，並提出解決此問題之對策。最後，初步的實驗結果顯示，成對式訓練方法RankNet之訓練模型檢索成效較逐點式訓練方法SVM之訓練模型檢索成效為佳。

Abstract

This thesis investigates the use of machine-learning approaches, namely learning-to-rank algorithms, for information retrieval (IR), with special emphasis on their theoretical foundations and the associated features that are used by them, such as the lexical features, proximity features, and probabilistic features. Meanwhile, we also consider the application of these approaches for spoken document retrieval (SDR). All experiments were conducted on the Topic Detection and Tracking corpora (especially, TDT-2 and TDT-3), which are the benchmark collections widely adopted for various SDR evaluations since they contain tens of hours of mainland-accented Chinese broadcast news documents equipped with topic labels and orthographic transcripts. In the hope of discovering more useful speech-related features for SDR as well as analyzing the problems caused by speech recognition errors, a large vocabulary speech recognition (LVCSR) system that can output a word lattice consisting of multiple recognition hypotheses for each broadcast news document is established. Moreover, we also deal with the problem of training the machine-learning retrieval models with unbalanced training data, and propose a remedy for it. Finally, the preliminary experimental results seem to show that the RankNet based retrieval model outperforms the support vector machine (SVM) based retrieval model for the SDR task studied in this thesis.

誌謝

終於要邁入下一個人生階段了，感謝我的爸爸、媽媽一路以來對我的支持跟鼓勵。謝謝您們養我、育我、教導我，您們是我心裡最大的支柱。謝謝您們在我成家以後，仍然辛苦的幫我帶孩子，讓我無後顧之憂。謝謝您們在我徬徨無助時，給予我最大的力量前進。謝謝您們對我的鞭策，讓我能夠完成我的學業。對您們的感謝，女兒只能用同樣的心孝順您們，用同樣的心養育我的下一代來盡些微薄的報答。

感謝我的指導教授—陳柏琳博士。感謝老師對我在學業上的教導，讓我比大學畢業時，更有信心面對我的未來，更有自信站上社會的舞台。感謝老師在我研究態度及為人處事的教導，我明白自己讓老師操了很多心，但老師的教誨都將銘記在我心中。謝謝我的三位口試委員—王新民博士、洪志偉博士及劉昭麟博士。謝謝老師非常仔細的幫我批改我的論文，提供論文意見，讓我的論文更臻完美。

我還要感謝實驗室的學長姐、學弟妹們。謝謝炫盛學長總是不厭其煩幫我解惑，學長的急時雨真的給我很大的幫助。謝謝庭瑋學姐對我一直以來的關心，妳的認真態度一直是我的模範。謝謝士弘學長、志豪學長、鴻彬學長對我的關心詢問。謝謝士翔學長及芳輝學長對我課業上的幫忙。接著，我要特別感謝鴻欣，不論在研究上或是人生道理上，我都受益良多。也要特別感謝實驗室的學弟妹們，永典、韋豪、冠宇、鈺玫、家玟，一直以來，實驗室都是學長姐們幫助學弟妹，唯獨我受你們的幫助更多，感謝你們一直給我支持的力量，在我沮喪時給我安慰，真的很謝謝你們。

最後，我要謝謝我的老公人瑋，和我的孩子力嘉。謝謝老公總是支持我、愛護我。謝謝我的力嘉，你的出生是媽媽幸福的最大來源，可以讓媽媽所有的煩惱都不見，謝謝你在媽媽趕論文時乖乖成長。我的家人，我真的好愛你們。

斯涵　謹誌

目錄

表目錄

圖目錄

1. 緒論

1.1 研究背景

資訊檢索(Information Retrieval, IR)是，從大量沒有結構化原始資料(通常為文字資料)所組成的語料庫(通常儲存於電腦)中，找到所需的資料，以滿足資訊需求(Information Need) [Manning et al. 2007]。資訊檢索的議題可以用自動資訊檢索系統(Automatic Information Retrieval System)來實作。所謂的自動資訊檢索系統中的「自動」，為使用電腦自動化並無人工介入，而在本論文中所使用「資訊」的單位則為文件(Document)。因此，本論文的研究重點是，如何以自動化的方式，將符合使用者資訊需求之文件準確地找出來。此外，在自動資訊檢索系統中，使用者會將資訊需求以查詢(Query)的方式表示；通常是一到數個詞，或者是一個句子，或一篇文件。根據查詢，我們可以發展各種檢索方法以搜尋出符合使用者需求的相關資訊。

資訊檢索的目的在於提供使用者更簡易的方法去取得所需要的資訊，而如何檢索出各種不同種類的資訊型態也越來越被受重視。因此，在這個被眾多資訊流淹沒、而且資訊量極度擴增的年代，資訊檢索儼然已經扮演著在網路世界中搜尋資料，一個不可或缺的媒介。甚至可以說，如果沒有一個好的檢索系統，網路上的眾多資訊似乎就無法獲得充份運用[Liu 2008]。正如一個很大的圖書館，卻沒有一個完善搜尋圖書的管道，縱使擁有再多的圖書資料，我們依然需要耗費龐大的時間才能夠獲得知識。所以，透過資訊檢索的平台，我們期望可以快速並且精準地搜尋出所需要的資訊，以增進知識的累積，並且節省獲得知識所需的時間。

如何讓資訊檢索的成效更為彰顯，一直是資訊檢索眾多研究者探討的重要課題。傳統的資訊檢索方法中，原則上可以歸類為兩個主要層面：比對策略(Matching Strategy)以及學習能力(Learning Capability) [Chen 2006]。

比對策略又經常使用兩個層面來判斷查詢與文件之間的相關程度

(Relevance)。此兩種層面分別為：逐詞比對(Literal Term Matching)及概念比對(Concept Matching) [Chen 2006]。架構在逐詞比對上的檢索方法相當多，最典型的例子為布爾模型(Boolean Model) [Baeza-Yates & Ribeiro-Neto 1999]。布爾模型使用：且(And)、或(Or)、不(Not)，將查詢以布爾(Boolean)表示式呈現，如果文件滿足查詢的布爾表示式，則文件就被認定為與查詢相關。另外，向量空間模型(Vector Space Model, VSM)[Salton 1968]亦是架構在逐詞比對上，但是較布爾模型更為複雜並且更具有調整的空間。向量空間模型，是將查詢與文件皆分別以向量表示之；向量中的每一維度代表語言中的某一特定詞之統計資訊。當分別完成了查詢及文件之對應向量表示式後，就可以利用餘弦評估(Cosine Measure)，對兩個向量進行相似度計算。當餘弦值越大，則代表查詢與文件越相關，反之則越不相關，本論文將在第三章對此模型將作更進一步之說明。概念比對則是發覺到，除了詞面上的相關度之外，其實不同的詞之間亦是有一些隱藏的關係。譬如：「美國」與「柯林頓」，雖然在詞面上無任何相關，但是此兩個詞彙實質上是有類似主題的。基於上述觀察所發展的方法最為有名的是隱藏語意索引(Latent Topic Indexing, LSI)：隱藏語意索引最主要的想法為，將每一個文件向量與查詢向量表示式投射到一個較低維度的空間之中，而這個低維度空間的每一維度可以和某些主題或概念作連結[Baeza-Yates & Ribeiro-Neto 1999]，如此一來，即使是在詞面上比對完全無關的模型，查詢與文件也可經由此種方法展現出彼此主題的相關性，提高檢索的成效。有關隱藏語意索引更深入的說明，亦將會呈現在本論文的第三章。

在[Salton & Buchley 1988]之研究中，曾針對向量空間模型查詢向量及文件向量中每一個詞作不同權重設定的討論，並歸納出權重設定之策略。由此，我們可以發現，在同樣的檢索模型下，對於不同種類的檢索語料，權重的給定方式常會不同。但通常需要耗費相當多的時間去嘗試，才能夠找到適合的權重設定。針對上述問題，我們可以引入機器學習方法，讓檢索模型擁有學習能力，也就是能自動化地得到適合的權重或模型參數設定。在具學習能力檢索模型之研究，有例如

隱藏式馬可夫模型(Hidden Markov Model, HMM)被應用於資訊檢索[Miller et al. 1999; Chen et al. 2004]上，它不僅是以機率角度出發的檢索模型，並且可以使用期望最大化演算法(Expectation-Maximization, EM)，對檢索模型進行訓練，因此找到有較好檢索效能的模型參數。大體而言，對於具學習能力之檢索模型研究上，不論是監督式(Supervised)或非監督式(Unsupervised)學習，都是希望使用各種機器學習(Machine Learning)方法及訓練方式，以期檢索系統達到更佳之成效。

資訊檢索領域中一樣會面臨到資訊安全的問題。非公開資訊檢索(Private Information Retrieval, PIR) [Chor et al. 1998]目的即是為了保障檢索資訊時的安全問題。有些資訊文件庫是相當敏感的，例如：股票、醫藥相關、個人資料等，進行檢索時，使用者當然相當希望能夠保障其檢索的隱蔽性，不會暴露以及紀錄檢索內容。最簡易的 PIR 作法為，使用者將所有的資料接收到使用端，在使用端檢索。因此使用者不需要傳送查詢給伺服器端，以此保護查詢的安全。

1.2 資訊檢索於多種資訊型態之應用

自有人類歷史以來，資訊就有各種不同的型態，隨著文明不斷地演進，資訊型態亦越見繁複，不同資訊型態的結合在今日亦是經常發生[陳光華 1999]。另一方面，由於電腦科技的蓬勃發展，網路傳送速度的提升與網路上各種活動的日益頻繁，可以被檢索的資訊可以有以下列幾種類型存在：

1. 純文字(Pure Text)：

過往在檢索的議題上，通常是以純文字檢索為主。初始的研究中著重在將查詢與文件中的字作比對或詞作比對，因此，許多探討詞重要性的議題隨之產生。例如在[Luhn 1958]中認為，文件中出現的詞頻率(Word Frequency)是一項非常重要的指標，可用於決定詞之重要性。而由於純文字文件牽涉到各種語言的特性，因此除了字比對與詞比對之外，其它各類自然語言處理(Natural Language Processing, NLP)的技術也被應用在純文字文件處理之中。自然語言處理的範疇

相當廣，而有被應用在資訊檢索議題上進行處理者，例如，句法分析(Parsing) [Keselj 1997]、詞性標註[Meteer et al. 1991]、自動摘要(Automatic Summarization) [Hardy et al. 2002]等，皆是利用更高階的自然語言處理技術對檢索文件進行分析，以得到更多除了詞比對之外的資訊，對於資訊檢索之成效亦會有幫助。

2. 圖像(Image)：

圖像檢索的研究主要可分為兩個時期。第一個時期：最初的圖像檢索研究盛行於 1970 年代時[Bashir 2002]，圖像檢索架構在純文字檢索的概念之上，先對圖像做圖義註解(textual annotation)。例如，在 Art and Architecture Thesaurus (AAT) 中提出了縱向 33 階層的類別以及橫向 7 個面向的圖像描述架構，就是為了鉅細靡遺地描述圖像[Goodrum 2000]，提供圖像的對應資訊。而使用者在進行檢索時，就是對這些已轉換為文字的資訊進行比對。這個時期的檢索方式有很大的缺點，其缺點在於文件資訊對圖像的解釋未必精確。例如：一幅畫標示為「裝著酒的杯子」，事實上可能這幅畫是跟「基督徒群聚」有關[Goodrum 2000]。除此之外，所有的圖像都必須經過標示，這樣的過程勢必耗費大量資源並且相當耗時。想要解決這樣的問題，於是開始了第二個時期的研究。第二時期的研究開始於 1990 年代，其著重在圖像內容檢索(Content-based Image Retrieval, CBIR)，方法為直接針對圖像本身，產生圖像原有的重要特徵。例如：顏色(Color)、形狀(Shape)、質地(Texture)、姿態(Motion)、及特別關聯的物件[Goodrum 2000; Bashir 2002]。以圖像內容檢索概念發展成功的搜尋引擎包括了 IBM 團隊的 QBIC[Flickner et al. 1995]。

3. 視訊(Video)：

視訊檢索和圖像檢索類似，但是視訊檢索又更為複雜，而視訊檢索的需求和圖像檢索亦類似[陳光華 1999]。在視訊檢索中一樣需要了解視訊內容，而以視訊內容為導向之檢索(Content-based Video Retrieval, CBVR)與下列四個過程有關：視訊內容分析(Video Content Analysis)、視訊結構語法剖析(Video Structure Parsing)、視訊摘要(Video Summarization)及視訊索引(Video Indexing) [Sebe et al.

2003]。視訊內容分析遇到的最大問題，在於無法輕易地將視覺化(Visual)的特徵對應到隱藏的語意概念(Semantic Concept)。我們可以很容易的得到顏色、形狀、結構等等的資訊；但卻很難從這些資訊中輕易得定義出實質的意義，像是影片中的人群正在喝酒。於是在視訊內容分析上，有時會引用一些其它的資訊來輔助了解，例如視訊的聲音資訊，視訊的文字資訊等等[Sebe et al. 2003]。視訊結構語法剖析是將視訊根據不同場景(Scene)進行切割的過程，在[Otsuji et al. 1991]中，即是利用不同框架(Frame)中，所有像素(Pixel)之顏色飽合度(Intensity)變化量來區別場景是否有所改變。視訊摘要則是在整段視訊擇選出最能夠代表全段視訊之部份段落，摘要的結果不僅能夠呈現視訊的內容，亦能夠幫助視訊檢索時對視訊內容的掌握及了解[Sebe et al. 2003]。視訊索引可以幫助視訊檢索，在對視訊進行分鏡(Shot)動作之後，對主要的鏡頭建立索引。而通常我們在進行檢索時，會給予關鍵詞(Keywords)，因此只要去比對關鍵字與建立好之索引關係，就能進行檢索。例如在[Petkovic et al. 2002]中曾探討視訊索引在網球比賽類型視訊中之應用。

4. 語音(Speech)：

在語音方面的檢索可以分為三種方式。

(1) 以文字查詢(Text Query)，檢索語音文件(Spoken Documents)。

(2) 以語音查詢(Spoken Query)，檢索文字文件(Text Documents)。

(3) 以語音查詢，檢索語音文件。

這三種方式皆各有其不同的應用之處。第一種可以應用在搜尋廣播新聞，收聽某一天的廣播新聞時，如果我們希望可以直接找出某一段新聞，不需要聽過整段節目，此時可以手動鍵入查詢，然後直接播放該新聞。第二種則可以應用在手機上，當我們不想要一個一個搜尋手機中所存放的電話簿時，我們可以用語音說話輸入人名，就可以直接找到該人名之電話。第三種則可以應用在，當我們身處在不適合鍵入資料、只適合輸入語音的環境；例如在開車行進間，希望能夠搜尋一些含語音資訊的多媒體來播放，就需要用到第三種檢索方式。由此

可見，語音檢索的議題可以有許多不同的面向做為探討。而不論是哪一種檢索類型，語音查詢與語音文件都必須透過語音辨識(Speech Recognition)技術，將語音型態資料轉換成以關鍵詞(Keywords)、音素串(Phone Strings)以及字串(Word Strings)所形成之內文特徵(Context Features)，如此一來，語音型態資料才有辦法被估量計算[Bai et al. 2000]。本論文將在下一小節中，更進一步探討語音文件檢索的整體流程及探討語音文件所面臨的議題。

面對以上如此多樣的資訊檢索目標群，我們仍希望不論是哪一種型態之文件，都能有很好的檢索成效。然而，面對不同的資訊檢索目標時，所使用的資訊檢索方法必然有所不同之處。本論文將會使用語音正確轉寫文件以及，經由 Dragon 語音辨識器、臺師大大陸口音中文大詞彙語音辨識器[Chen et al. 2004, 2005]，各別轉寫之語音文件進行資訊檢索。Dragon 語音辨識器轉寫結果為 TDT 語料所提供，而選用臺師大大陸口音中文大詞彙語音辨識器轉寫則是希望探討，在較艱難辨識環境（缺乏語料之詞典及語料之語言模型）下辨識之結果（正確率較低）的檢索成效。我們期望能夠應用機器學習方法於語音自動轉寫文件之中，提升語音自動轉寫文件的檢索能力。

1.3 語音文件搜尋研究之介紹

語音文件在現實的環境中是廣佈在我們的週遭的，例如 CNN TV 廣播新聞播放全球新聞、MIT 開放式課程錄音等文件。而語音長久以來是人與人之間一項主要並且最為便利的溝通方式。並且隨著科技發展，電子設備的體積越來越小，還有無線通訊及網絡的蓬勃發展之下，我們可以相當期待語音在不久的未來世界當中，不僅在人與人之間扮演著重要的溝通橋樑，甚至也將在人與機器之間做為重要的溝通媒介[Chen 2006]。當前語音文件搜尋研究主要有兩大主軸，分別為口說詞偵測(Spoken Term Detection, STD)和語音文件檢索(Spoken Document Retrieval, SDR) [Meng et al. 2007]。以下將分別對這兩大研究主軸進行介紹。

口說詞偵測又可稱為關鍵字擷取(Keyword Spotting) [Meng et al. 2007]，主要的目標是在給定的語音語料中找出某些特定的詞(Word)或詞組(Phrase)所有出現的位置。在口說詞偵測研究之中系統通常須使用者輸入文字查詢(通常是一至三字詞)[NIST 2006]，將查詢與語音文件的自動轉寫進行詞層次(Word-level)或次層次(Subword-level)比對[Meng et al. 2007]，找出含有部份或全部文字查詢詞的語音文件。口說詞偵測強調的是逐詞比對(Literal Term Matching)，而不強調主題或概念相關的搜尋。例如輸入查詢是「歐巴馬」，就只能搜尋出含有「歐巴馬」一詞的語音文件，若文件中僅有「美國總統當選人」或者是歐巴馬的演講內容，則是無法被偵測出來的。

回溯在 1996 年間，Spärck Jones 探討如何將既有的資訊檢索方法應用在語音辨識結果之中[Garofolo et al. 2000]，開創了語音文件檢索研究的先端。語音文件檢索是自動語音辨識技術與資訊檢索技術的結合[Garofolo et al. 2000]。自動化語音辨識過程為：一個給定音訊串流(Audio Stream)，在經過自動化語音辨識器(Automatic Speech Recognition, ASR)之後，可以得到一個具有時間標籤的語音轉寫(Speech Transcript)。此轉寫結果的表現方式可以是音素(Phone)圖或詞(Word)圖[Ortmanns et al. 1997]，並且，音素圖或詞圖也可以用兩種方式來作為呈現，第一：可以選擇找出一組結果最好的轉寫結果，例如機率值最高者為最好的轉寫結果；或者，第二：在每一個單位時間點上皆找出前 N 組最佳的音素或詞，使語音檢索文件擁有更多種可能的轉寫結果提供選擇[Garofolo et al. 2000]。語音文件檢索旨在搜尋主題上相關的文件(Topic-Relevant Documents)以回應使用者輸入的口說或文字查詢(Spoken or Text Queries)。語音文件檢索的查詢，可以是簡短的幾個詞，或是一篇文字或語音文件範例，亦即所謂的 Query-By-Example。大詞彙連續語音辨識(Large Vocabulary Continuous Speech Recognition, LVCSR)技術常被用來產生語音查詢與文件的自動轉寫，而檢索模型則是使用一般已常在文字檢索(Text Retrieval)使用的現有模型。近幾年來，隨著語音辯識技術的快速發展，在語音辨識上已經有重大的貢獻以及相當令人振奮的結果[Zhou et al. 2006]，而

正確率高的辨識結果將有助於檢索成效的提升。

文字文件檢索的研究中，經常使用詞出現的頻率(Term Frequency)、是否為虛詞(Function Words)或者是否為關鍵詞等資訊，融入檢索函數中，據此對文件進行排名。在語音文件檢索中，主要會使用到兩種方法[Mamou et al. 2006]，第一，根據查詢，找尋文件中語音學上相同的音素串(Phone Sequence)[Clements et al. 2002]，第二，需先將語音資訊透過大詞彙連續語音辨識器轉寫為文字資料，接著運用檢索文字文件的檢索模型進行檢索[Garofolo et al. 2000]。語音文件檢索受到自動語音辨識的影響甚大[Mamou et al. 2006]。因為，在自動轉寫的過程之中，如果發生轉寫錯誤，造成文件中有辨識錯誤的資訊，這些錯誤的資訊若是沒有經過處理，直接使用純文字檢索的模型進行檢索，會使得檢索的成效降低。過去，NIST(National Institute of Standards and Technology) SDR (Spoken Document Retrieval, SDR)Track 計畫引領了大部份的語音文件檢索研究。NIST SDR Track 著重在檢索廣播新聞語音語料庫的自動轉寫[Mamou et al. 2006]，計畫執行者曾對語音文件檢索研究下了一些結論，其中一個重要結論是—資訊檢索的成效大部份取決於轉寫文字的正確率。而根據 NIST SDR Track 報告顯示，參與單位的最佳語音辨識正確率已大於 90%；因此，NIST SDR Track 認為 SDR 是一個已解決之問題[Mamou et al. 2006]。然則，NIST SDR Track 並非將資訊檢索技術應用在自發性語音(Spontaneous Speech)辨識上[Meng et al. 2007]，而在自發性語音的語音辨識正確率並無法達到像廣播新聞如此高的效果[Saraclar et al. 2004]。而且，即使是最佳的辨識器，目前仍舊無法達到百分之百的辨識率。所以，在語音文件檢索的研究中，仍需要去探討如何提升在無法達到百分之百辨識率情況下檢索成效所將會受到的影響。並且，如果能夠找到比文字資訊更有代表性的特徵，那麼，語音文件將擁有文字文件未見之資訊，則語音文件檢索成效的最大臨界值，將不見得僅是文字檢索成效。

本論文將著重於搜集語音文件的特徵，並考量語音文件的獨特性，將其用於機器學習方式的檢索模型，以期不同的查詢皆能給予對應的相關(Relevant)語音

文件有較正確的排序結果。

1.4 本論文研究內容與貢獻

本論文初步討論機器學習之方法在資訊檢索上的應用，即所謂排序學習(Learning to Rank)。近幾年間，排序學習運用於資訊檢索上，是非常熱門的議題，我們亦將在第二章中深入介紹。論文中，針對近年被使用在資訊檢索上的各種機器學習模型及概念，並引進傳統資訊檢索方法做為各種特徵，包含詞彙本身之特徵、相關度特徵、及機率特徵等，進行分析與實驗。本論文的貢獻有以下幾點：

1. 本論文將排序學習延伸至語音文件檢索的應用上，使用具有錯誤訊息的文件，了解經過訓練後，是否能對語音文件有所幫助。
2. 我們實作出排序網路(RankNet)演算法[Burges 2005]，在使用兩層的神經網路模型時，對檢索效能有所提升。
3. 在語音文件檢索上，除了使用原語料 Dragon 語音辨識器自動轉寫之語音文件外，本論文初步地使用臺師大大陸口音大詞彙語音辨識器(Large Vocabulary Continuous Speech Recognition, LVCSR)轉寫產生之詞圖來擷取語音文件的特徵。
4. 我們使用排序學習中逐點式訓練之支援向量機(SVM)對語音文件進行訓練，雖然逐點式訓練的成效較不好，但我們仔細分析實驗之結果，探討其原因。
5. 我們初步探討如何對訓練語料進行篩選，改善資訊檢索中訓練語料的不平衡問題，雖然實驗之結果並不理想，但我們提出之改善方法，仍可提供作為未來訓練語料選取問題之參考。

1.5 研究內容架購

本論文接下來的安排如下：第二章將對排序學習於資訊檢索上之議題進行回顧，

並且針對排序學習(Learning to Rank)方法中訓練方法之一的支援向量機(Support Vector Machine, SVM)介紹。第三章將介紹本論文對語音正確轉寫文件擷取之特徵，利用支援向量機進行訓練，以及初步實驗設定與結果。第四章中，對於初步實驗之觀察後，提出兩種解決對策，一則以排序網路(RankNet)進行訓練，二則對訓練語料不平衡問題進行改良。第五章介紹語音文件檢索過程，以及大詞彙語音辨識器轉寫之語音文件產生過程介紹。第六章呈現語音正確轉寫、Dragon 語音辨識器轉寫之語音文件及臺師大大陸口音中文大詞彙語音辨識器轉寫語音文件，使用排序網路以及解決訓練語料不平衡問題實驗之結果。第七章中為本論文對整體實驗之結論。第八章為未來展望。第九章則是相關參考資料。

2. 文獻探討

2.1 排序學習(Learning to Rank)

近幾年來，排序學習不論是在機器學習議題上或是資訊檢索議題上，都非常活躍且快速發展。排序學習的主要目地是希望能夠根據已經有正確排序的訓練語料中，自動地訓練出一個排序模型，使得此模型可以用於排序或者物件分類 (例如，文件)。資訊檢索也可以被視為是一種排序問題，例如，給定一個查詢，而檢索系統根據查詢計算所有文件各別與查詢相關分數(Relevance Score)。最後依照相關分數大小作排序，相關分數越大者代表相關度越大[Xu et al. 2006]。

然則，在排序學習應用於資訊檢索之前，資訊檢索的監督式學習議題早已經誕生。在 1970 年間，資訊檢索的問題建立在一篇文件對應一個查詢，有相關(Relevant)與不相關(Non-relevant)兩種狀態之假設上。監督式學習在此假設下發展的檢索模型：二元不相依檢索(Binary Independence Retrieval, BIR)模型[Roberson et al. 1976]、Rocchio's 回饋演算法[Rocchio 1971]。其中，二元不相依檢索模型可將資訊檢索問題視為一種分類問題[Nallapati 2004]。其將相關類別(Relevance Class)以 R 表示，不相關類別以 $\overline{R}$ (Irrelevance Class)表示。接著將一篇文件 d 轉換為一個 n 個特徵值之向量 $d=(x_1,x_2,...,x_n)$，n 是存在於詞典中詞的總數，x_i 是一個二元隨機變數，當詞典中第 i 個詞有出現於 d 中，則表示為 1，反之為 0。選用對數相似度(Log-likelihood Ratio)做為排序之估計值。此對數相似度根據貝氏定理(Bayes' Rule)可以做以下之轉換：

$$\log(\frac{P(R\mid d)}{P(\overline{R}\mid d)})=\log(\frac{P(d\mid R)}{P(d\mid \overline{R})}\frac{P(R)}{P(\overline{R})}) \tag{2.1.1}$$

假定向量中之每一個維度(詞)彼此之間皆獨立，式(2.1.1)可以轉換為：

$$\log(\prod_{i:x_i=1}\frac{P(x_i=1\mid R)}{P(x_i=1\mid \overline{R})}\prod_{i:x_i=0}\frac{P(x_i=0\mid R)}{P(x_i=0\mid \overline{R})}\frac{P(R)}{P(\overline{R})}) \tag{2.1.2}$$

第一次進行估算時，我們無法真正得知 R 中為哪些文件的集合，因此，使用一致

(Uniform)機率。之後，則可根據使用者相關回饋(User Relevance Feedback)，提供 *R* 資訊，便可重新估算對數相似度並給予排序。這是初期納入使用者提供之資訊所進行之監督式學習。

資訊檢索問題視為一種分類問題(Classification Problem)時，就可以很容易的利用機器學習進行訓練。在 1980 至 1990 年代間，[Drucker et al. 1999; Joachims 1998] 將機器學習方法之一的支援向量機(Support Vector Machine, SVM)，應用於文件分類(Text Categorization)。將每一個訓練語料中之文件擷取若干特徵，並給予文件正確標籤(Labels)。對已有文件標籤之訓練語料進行訓練，可得到一個分類模型。需被分類之文件，只要透過此分類模型，即可以被指定至某一類別。這些被分類文件雖說並沒有被用於資訊檢索，但是，相關文件分類方法與概念不斷被延伸與改進。在 2001 年[Berger 2001]說明使用訓練語料的訓練查詢與文件配對，並對訓練查詢與文件對給予正確相關或非相關標籤。對給予標籤之訓練語料進行訓練，得到分類模型。在 2004 年，Ramesh Nallapati 在 SIGIR，正式介紹使用訓練查詢與文件配對，透用最大熵值(Maximum Entropy)、支援向量機之鑑別式模型(Discriminative Models)應用於資訊檢索訓練上[Nallapati 2004]。而在 2005 年，NIPS(Neural Information Processing Systems)特別針對排序學習應用於資訊檢索，舉辦了研討會。2007 年 SIGIR 也特別地另外舉辦排序學習研討會，2007 年及 2008 年，SIGIR 已將排序學習正式獨立為一個議程，目的在討論當前排序學習之相關議題。由此可見，排序學習在資訊檢索上的議題受到相當大的重視。

排序學習為一種監督式學習，特別是在向量空間上進行學習[Liu 2008]。近年來排序學習在資訊檢索的應用，大致上可以分為三個方向: 逐點式訓練(Point-wise Training)、成對式訓練(Pair-wise Training)、序列式訓練(List-wise Training)。圖 2.1 展示了這三種訓練方式在資訊檢索研究之演進。以下將分別介紹這三種訓練方法之概念。

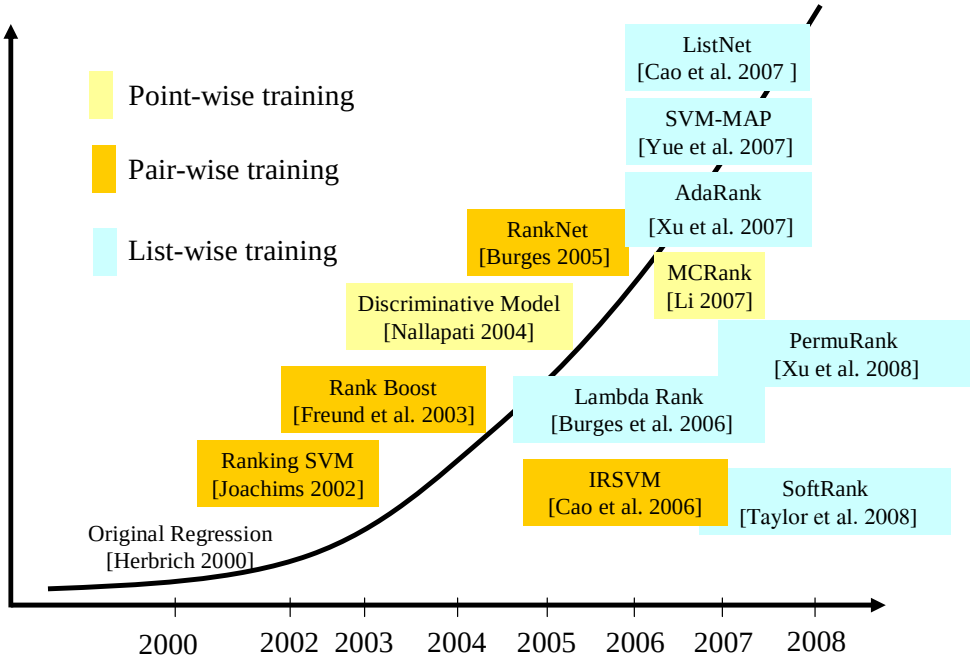

圖 2.1、逐點式訓練、成對式訓練及序列式訓練在資訊檢索議題下之演進

2.1.1 逐點式訓練(Point-wise Training)

逐點式訓練是將排序的問題轉換成求取迴歸線(Regression)或者看做是一種分類(Classification)的問題。在 2004 年，Nallapati 發表了將鑑別式模型應用在資訊檢索上[Nallapati 2004]，就是一種逐點式訓練。Nallapati 闡述資訊檢索的議題為二元分類問題；亦即，有「相關」及「不相關」兩個類別。他將一個訓練查詢與一個文件配對為一組例子(Instance)，若是此查詢與文件為相關，則應分類至相關類別(使用+1 做為標籤)，若為不相關，則應分類至不相關類別(使用-1 做為標籤)。訓練語料中的訓練查詢對應至所有的文件，皆可以計算出各種特徵，而 Nallapati 總共定義了 6 個特徵。一個訓練查詢與一個文件可以用一組向量來表示，而此向量即是由 6 個特徵所組成。而每一個向量，都可以使用訓練語料中已知的對應關係來各自標記。接著，他利用支援向量機，對所有訓練語料所在之向量空間做訓練，產生分類模型。當測試資料須要分類時，即可透過分類模型進行分類。最後，將分類之結果進行排序，分類為相關之文件，其排序應在不相關之文件之前。除

了 Nallapati 提出的方法之外，2007 年 Li 發表在 NIPS[Li 2007]，提出了不只分為兩類的逐點式訓練法，他將訓練語料分為 5 種相關程度，並且使用 Gradient Boosting 進行分類的動作。圖 2.2 為逐點式訓練的實驗示意圖[Liu 2008]。

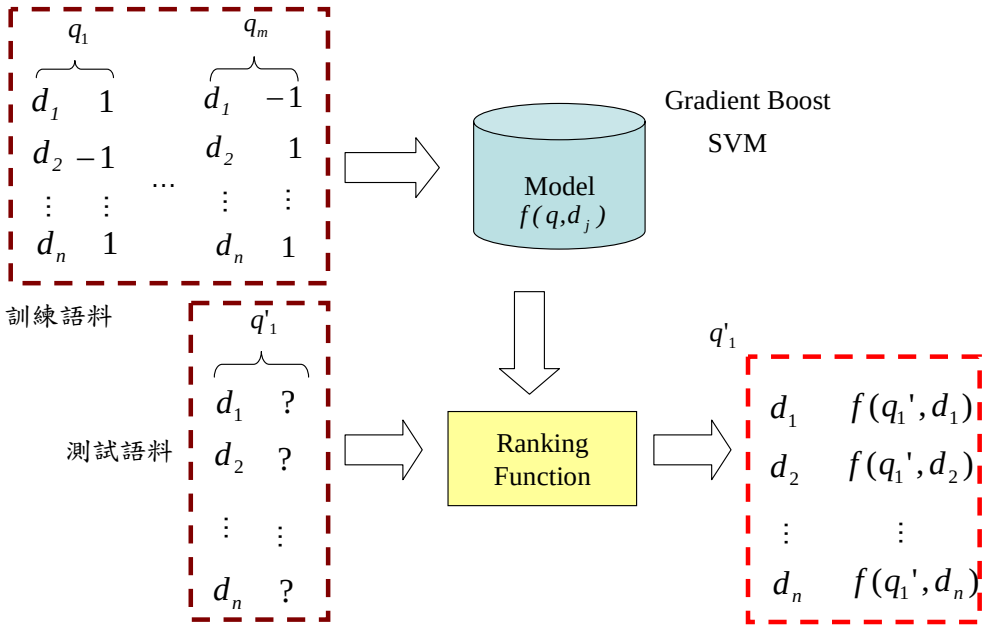

圖 2.2 逐點式訓練之示意圖

訓練語料中，對一「查詢/文件對組」進行標籤，可透過支援向量機或 Gradient Boosting 進行訓練。得到訓練模型之後，測試查詢與文件組成之向量可透過此模型進行分類，再經過排序系統，對分類結果進行排序(相關文件之排名應在不相關文件之前)，得到排序結果即為檢索結果。

2.1.2 成對式訓練(Pair-wise Training)

成對式訓練可視為一種分類、迴歸或是有序性迴歸(Ordinal Regression)的問題。Salton 中提出，當給定一查詢，其對應的文件關係，應為多層次的相關性[Salton 1968]；亦即，應不只有相關、不相關兩種類別，而是應有高度相關、較相關、較不相關、低度相關等區別。因此，他首先開始發展出了如何去訓練有層級性相

關度的排序。成對式的機器學習訓練早在 1995 年就已被提出[Caruana 1995]，當時以醫療診斷的資料庫做實驗，可得到相當不錯的成效。在 2000 年[Herbrich et al. 2000]提出有序性迴歸之概念，2002 年，Joachims 運用有序性迴歸提出了 Ranking SVM [Joachims 2002]，並將有序性迴歸首度應用於資訊檢索上。成對式訓練主旨是針對某一個查詢，考慮兩兩成對的文件之關係；亦即，對於查詢 q，同時考慮 d_1 與 d_2 的排序關係。當實際上 d_1 的排序順序高於 d_2，則 $d_1 \succ d_2$，反之則為 $d_2 \succ d_1$，其中 $d_1 \succ d_2$ 其英文含意為「d_1 is preferred to d_2」。對於每一文件，都有一個特徵向量及給定之排序標記值，對於任一文件 d_i，$d_i = (\vec{x}_i, y_i)$，同樣的，對於文件 d_j，$d_j = (\vec{x}_j, y_j)$，我們可以合併這兩個文件的特徵向量，形成 $\vec{x}_i - \vec{x}_j$，於是我們可以將訓練資料轉換以下列形式表示:

$$\left(\vec{x}_i - \vec{x}_j, z = \begin{cases} +1 \text{ if } y_i \succ y_j \\ -1 \text{ if } y_j \succ y_i \end{cases} \right)$$

而成對式訓練之示意圖如圖 2.3 所示。

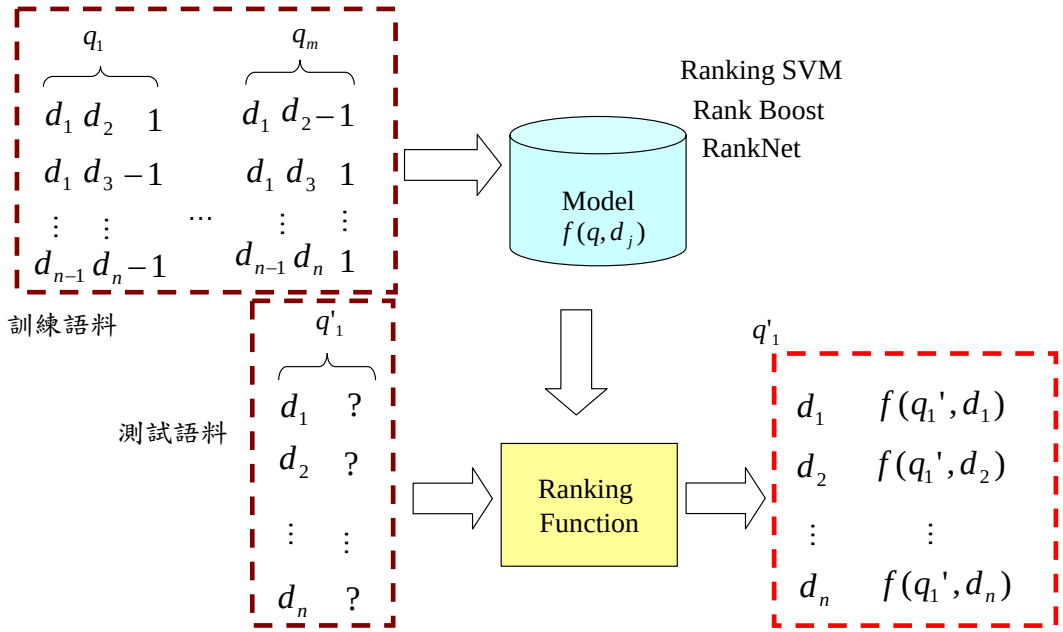

圖 2.3 成對式訓練的實驗示意圖

訓練語料中，一個訓練查詢對應之兩兩文件對根據訓練語料分別給予的標籤，重新標籤。若在訓練查詢q中，d_i排序在d_j之前，則$d_i - d_j$標示為$+1$，反之則標示為-1。標記完之後查詢及所有文件對，可以透過 Ranking SVM 進行訓練，得到一個訓練模型。最後，測試語料經由此訓練模型可得到標籤值，可作為排序依據，最後排序結果即可視為檢索結果。在資訊檢索排序學習的研究領域中，目前著名的成對式訓練方法及分類器除了 Ranking SVM 外，還有 RankBoost [Freund et al. 2003]及 RankNet [Burges 2005]等。

2.1.3 序列式訓練(List-wise Training)

序列式訓練是最近幾年才被提出的方法，在 2007 年的 SIGIR 有兩篇論文同時被發表：一篇由 Yue 等人提出的 SVM-MAP [Yue et al. 2007]，為 Ranking SVM 的延伸方法，直接針對平均精確率作檢索模型最佳化；另一篇是由 Xu 等人提出的 AdaRank [Xu et al. 2007]，為 RankBoost 的延伸，讓檢索模型可對平均精確率(Mean Average Precision, MAP)及均化遞減累積效益(Normalized Discount Cumulative Gain, NDCG)作最佳化。在 2007 年 ICML 會議上，Cao 等人也將 RankNet 延伸到 List-wise Training，為著名的 ListNet [Cao et al. 2007]。成對式訓練與序列式訓練的最小錯誤函數都是以最小排序文件錯誤做為估算，當每一序列只含兩個文件時，序列式訓練即退化為成對式訓練，故成對式訓練可視為序列式訓練的一個特例。另一方面，一個序列亦可拆成有限個文件對組(Pairs)，故成對式訓練基本上已能擷取到大部份的排序資訊，因此也可說明為何序列式訓練在資訊檢索實際上相較於成對式訓練在效能增進上沒有辦法獲得很大的躍進。

2.2 支援向量機(Support Vector Machine)

由於，排序學習中之逐點式學習可運用支援向量機進行訓練。因此，以下我們將先對支援向量機作介紹。支援向量機是一種最大邊界(Large-Margin)的分類器。

由 Vapnik 及其在 AT&T Bell 實驗室之團隊所共同發展之技術[Boser et al. 1992; Cortes & Vapnik 1995]。

在 n 維向量空間之中，屬於不同類別(Classes)之資料點可以用 n 維的向量表示。例如在資訊檢索研究之中，單位資料點就是查詢與某篇文件(Documents)所組成的向量；不同的類別則可以分為與查詢相關(Relevant)類別或是與查詢不相關(Non-relevant)類別；每一維度都可以用一特定之特徵來表示，n 個維度代表有 n 個特定特徵。支援向量機在資料點散布之 n 維向量空間之中，尋找決定函數(Decision Function)。以及邊界(Margin)值。決定函數為一個超平面(Hyper-plane)，決定函數切割 n 維空間，使所有的資料點由決定函數區分為不同類別。當類別為多個類別時，決定函數可使用一對多(One-against-All) [Vapnik 1995]或一對一(One-against-One)的方法來決定，為了便於了解支援向量機，以下只考慮兩個類別，不考慮一對多之情形。以兩個類別來說，分別在兩類別中，所有距離最接近決定函數所形成超平面之數個資料點，稱之為支援向量(Support Vectors)，而此超平面至分屬不同類別之支援向量的距離和，則稱之為邊界。如圖 2.4 所示，在兩個類別的問題中，$f(\vec{x})$ 為決定函數，決定函數至兩類別的支援向量之距離為邊界。

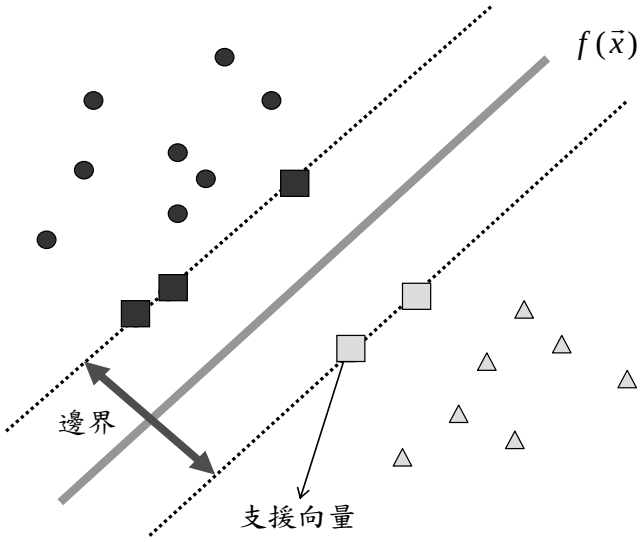

圖 2.4 支援向量機示意圖(1)

支援向量機的目標在於找到最大之邊界值。位於決定函數兩端的支援向量決定了邊界值的距離。而支援向量的取得，是由原先散布在 n 維空間中的資料點中去選定。因此，支援向量機的目標，則是在所有的資料點中，找到最適當的支援向量，用以得到最大之邊界值。我們期望，未來在支援向量兩端所畫分的空間中，沒有其它的資料點存在；如此以來，就代表這些支援向量可以清楚地畫分整個向量空間，將新的資料點正確地劃分為兩個類別。

以下，我們將假設在 n 維空間中的資料點是用線性可分割(Linearly Separable)，如圖 2.5 所示。

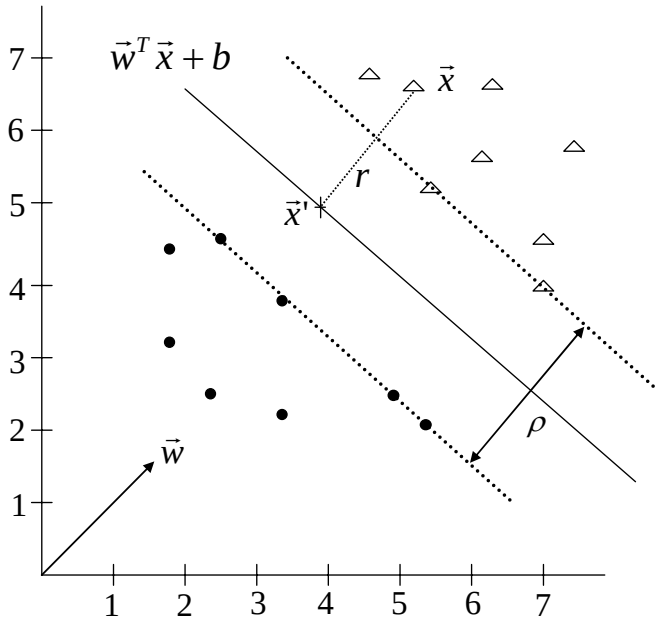

圖 2.5 支援向量機示意圖(2)

並且，空間中的所有資料點皆以式(2.2.1)表示，其中，$\vec{x}_i$ 代表資料點，y_i 代表此資料點之類別標籤(Label)，例如+1 或者-1。圖 2.5 中，我們假設某決定函數為 $f(\vec{x})=\vec{w}^T\vec{x}+b$，而 $\vec{w}$ 為此決定函數之法向量(Normal Vector)。在 $f(\vec{x})$ 所形成之超平面上，在所有資料點中任取一點 $\vec{x}$，在超平面上且與 $\vec{x}$ 最近之距離點為 $\vec{x}'$，因此，我們可以估算 $\vec{x}$ 至 $\vec{x}'$ 之距離 r，即為 $\vec{x}$ 至超平面之距離。由式(2.2.2)，我

們知道$\vec{x}$為$\vec{x}'$加上一個與$\vec{x}'$同方向且長度為r之向量，經過移項，便可以得到$\vec{x}'$之表示式(2.2.3)。接著，將$\vec{x}'$引入$f(\vec{x})$之中，我們可由式(2.2.4)得到距離r。式(2.2.4)中，可以得知，不論$f(\vec{x})$或是將$f(\vec{x})$乘以某倍數之函數，皆不會影響此r之距離，舉例而言，即使$f(\vec{x})=\vec{w}^T\vec{x}+b$與$f'(\vec{x})=5\vec{w}^T\vec{x}+5b$為不同之函數，但在式(2.2.4)中仍會得到相同之結果。因此，我們只以 1 倍率做為考量。

$$D=\{\vec{x}_i,y_i\}_{i=1}^{N} \tag{2.2.1}$$

$$\vec{x}=\vec{x}'+r\cdot y\cdot\frac{\vec{w}}{|\vec{w}|} \tag{2.2.2}$$

$$\vec{x}'=\vec{x}-r\cdot y\cdot\frac{\vec{w}}{|\vec{w}|} \tag{2.2.3}$$

$$\begin{aligned}&\vec{w}^T(\vec{x}-r\cdot y\cdot\frac{\vec{w}}{|\vec{w}|})+b=0\\&\Rightarrow r=\frac{y(\vec{w}^T\vec{x}+b)}{|\vec{w}|}\ \text{或}\ r=\frac{|\vec{w}^T\vec{x}+b|}{|\vec{w}|}\end{aligned} \tag{2.2.4}$$

當以上所估量之資料點$\vec{x}$為一支援向量時，距離r的兩倍即為我們所希望得到的邊界(Margin)值。當資料點$\vec{x}$為一支援向量時，我們稱$y(\vec{w}^T\vec{x}+b)$為函數的邊界(Functional Margin)，我們可以設定函數的邊界為 1，亦即$y(\vec{w}^T\vec{x}+b)=1$(不論設定為何，都不會影響到最後邊界之結果。因此，我們給予最簡便之值：1)。而其它非支援向量之資料點，則為$y(\vec{w}^T\vec{x}+b)\geq 1$。至此，我們可以得知期望得到之邊界值為$\frac{2}{|\vec{w}|}$。由此可知，若想要求得最大之邊界值，即是對$\frac{2}{|\vec{w}|}$做最大化之動作。對$\frac{2}{|\vec{w}|}$做最大化之動作即是對$\frac{|\vec{w}|^2}{2}$最最小化。而$\frac{|\vec{w}|^2}{2}$亦可以$\frac{\vec{w}^T\vec{w}}{2}$表示之。因此，求支援向量機之目標：最大邊界值，即為式(2.2.5)。

$$\begin{aligned}&\underset{\vec{w},b}{Minimize}\quad \Phi(\vec{w})=\frac{\vec{w}^T\vec{w}}{2}\\&subject\ to\\&\qquad y\cdot(\vec{w}^T\vec{w}+b)\geq 1\end{aligned} \tag{2.2.5}$$

式(2.2.5)為一個二次方程問題(Quadratic Problem)，理論上是可以被直接求解的

[Boser et al. 1992]。然而，當資料點所在之空間的維度非常的大甚或是無限時，這個問題其實很難求解。而通常，我們又都會在高維度上求解此問題[Boser et al. 1992]。所以，我們必需將式(2.2.5)利用其它的技術進行求解的動作。首先引入 Lagrange Multiplier 的技術，因此，我們可以得到一個 Lagrangian [Luenberger 1984]：

$$L_p[\vec{w},b,\Lambda]=\frac{1}{2}(\vec{w}^T\vec{w})-\sum_{i=1}^{N}\alpha_i[y_i(\vec{w}^T\vec{x}_i+b)-1] \tag{2.2.6}$$

其中 $\Lambda=(\alpha_1,...,\alpha_N)$ 為一個對應於式(2.2.5)中每一個條件，皆不為負值(Non-negtive)的 Lagrange Multipliers 之向量。式(2.2.6)之最佳解，即為對此 Lagrangian 求鞍點(Saddle Point)，此鞍點將會最小化 $\frac{\vec{w}^T\vec{w}}{2}$ 以及遵守 Λ 為非負向量之條件下將 $[y_i(\vec{w}^T\vec{x}_i+b)-1]$ 最大化。鞍點的求法如下：

(1) 對 $\vec{w}$ 偏微分：

$$\begin{aligned}&\frac{\partial L(\vec{w},b,\Lambda)}{\partial\vec{w}}=0\\&\Rightarrow\vec{w}=\sum_{i=1}^{N}\alpha_i y_i\vec{x}_i\end{aligned} \tag{2.2.7}$$

(2) 對 b 偏微分：

$$\begin{aligned}&\frac{\partial L(\vec{w},b,\Lambda)}{\partial b}=0\\&\Rightarrow\sum_{i=1}^{N}\alpha_i y_i=0\end{aligned} \tag{2.2.8}$$

(3) 將式(2.2.7)與式(2.2.8)引入式(2.2.6)：

先對式(2.2.6)進行拆解

$$\begin{aligned}\text{式(2.2.6)}&=\frac{1}{2}(\vec{w}^T\vec{w})-\sum_{i=1}^{N}\alpha_i y_i(\vec{w}^T\vec{x}_i+b)+\sum_{i=1}^{N}\alpha_i\\&=\frac{1}{2}(\vec{w}^T\vec{w})-\vec{w}^T\sum_{i=1}^{N}\alpha_i y_i\vec{x}_i-b\sum_{i=1}^{N}\alpha_i y_i+\sum_{i=1}^{N}\alpha_i\end{aligned}$$

將式(2.2.7)與式(2.2.8)，引入

$$
\begin{aligned}
L_d(\Lambda) &= \frac{1}{2}(\vec{w}^T\vec{w}) - \vec{w}^T\sum_{i=1}^{N}\alpha_i y_i \vec{x}_i - b\sum_{i=1}^{N}\alpha_i y_i + \sum_{i=1}^{N}\alpha_i \\
&= \frac{1}{2}(\vec{w}^T\vec{w}) - \vec{w}^T\cdot\vec{w} - b\cdot 0 + \sum_{i=1}^{N}\alpha_i \\
&= -\frac{1}{2}\vec{w}^T\vec{w} + \sum_{i=1}^{N}\alpha_i \\
&= -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j \vec{x}_i^T\vec{x}_j + \sum_{i=1}^{N}\alpha_i
\end{aligned}
\tag{2.2.9}
$$

為了將來也能在非線性(Non-linear)分割上亦能較容易求得最佳解，因此引入 KKT 條件(Karush–Kühn–Tucker Conditions)，將式(2.2.9)以矩陣形式標記，並且合併 Λ 的非負性質以及式(2.2.8)之條件，就可以轉換為一個雙重問題(Dual Problem)的形式：

$$
\begin{aligned}
&\textit{Maximize}\quad L_d(\Lambda) = \Lambda\cdot 1 - \frac{1}{2}\Lambda\cdot D\cdot\Lambda \\
&\textit{subject to} \\
&\qquad \Lambda\cdot\mathbf{y} = 0 \\
&\qquad \Lambda\text{為非負向量}
\end{aligned}
\tag{2.2.10}
$$

其中，$\mathbf{y} = (y_1,...,y_N)$，而 D 是一個對稱 $N\times N$ 矩陣，D 中每一個 $D_{ij} = y_i y_j \vec{x}_i^T\vec{x}_j$。而 KKT 條件之其中一必要條件：Complementary Slackness Condition，可以標為以下形式：

$$
\alpha_i[y_i(\vec{w}^T\vec{x}_i + b) - 1] = 0 \qquad i = 1,...,N \tag{2.2.11}
$$

在式(2.2.11)中，我們可以得知，只有當 $\alpha_i > 0$ 時，$[y_i(\vec{w}^T\vec{x}_i + b) - 1]$ 才會有所影響，而 $[y_i(\vec{w}^T\vec{x}_i + b) - 1]$ 即是式(2.2.5)之條件，亦即，只有當 $\alpha_i > 0$ 時，此資料點 $\vec{x}_i$ 才會有決定性的影響。而這些有決定性影響的 $\vec{x}_i$ 即是支援向量。

最後，當我們找到最佳之一組 $\vec{w}^*$、b^*及α^*我們可以重寫決定函數為：

$$
f(\vec{x}) = sign(\sum_{i=1}^{N} y_i\alpha^*_i(\vec{x}\cdot\vec{x}_i) + b^*) \tag{2.2.12}
$$

其中，$\vec{x}$ 為一新資料點，透過式(2.2.12)可以對此新資料點做分類，給定一個類別。

3. 資訊檢索架構與問題論述

傳統資訊檢索之模型，如表 3.1 所示：

模型分類	範例	參考
Proximity Models	向量空間模型 (Vector Space Model)	[Salton 1988]
	隱藏語意索引模型 (Latent Topic Indexing)	[Furnas et al. 1988]
	布爾模型 (Boolean Model)	[Baeza-Yates 1999]
Probabilistic Models	語言模型 (Language Model)	[Zhai et al. 2001]
	最佳配對權重 (BM25)	[Roberson 1995]
	隱藏馬可夫模型 (Hidden Markov Model)	[Miller et al. 1999]
Graph-based Models	HITS (Hyperlink-Induce Topic Search)	[Kleinberg 1999]
	PageRank	[Brin 1998]

表 3.1 傳統資訊檢索之模型

其中，相近度模型(Proximity Models)將資訊檢索問題視為一種計算查詢與文件之相關度之問題；機率模型(Probabilistic Models)將文件用相關機率做排序；Graph-based Models 利用文件間相關性或是連結統計資訊進行排序之模型。這些傳統資訊檢索方法，可以歸納出兩種問題。第一：對於單一特定模型，參數難以調整。如 BM25[Roberson et al. 1995]中有兩個參數需要調整，但是最佳的權重及參數並沒有一個具有理論基礎的計算方式。第二：歷年來雖有許多的檢索模型，但並沒有一個顯而易見的方法可以去結合這些模型。因此，使用機器學習，可以對這兩種問題做有效的解決。如何找到一個更有效的排序函數(Ranking Function)仍然是傳統資訊檢索與使用機器學習方法共通面對的問題。以下，將使用排序學習的逐點式訓練(Point-wise Training)，並選用其中的支援向量機(SVM)對訓練語

料進行訓練，利用訓練後之模型進行檢索，觀察其檢索效能。

3.1 Learning to Rank 在資訊檢索上的方法

在排序學習的逐點式訓練中，我們可將訓練語料中之查詢與文件各自組成一對組，成為「查詢/文件對組」。同時對每一組「查詢/文件對組」進行特徵選取。特徵選取之後，對所有的「查詢/文件對組」賦予經由訓練語料已提供之正確標籤：若查詢與文件為相關，則「查詢/文件對組」給予相關標籤；反之，若查詢與文件為不相關，則「查詢/文件對組」給予不相關標籤。

對所有「查詢/文件對組」進行標籤動作之後，即可透過支援向量機進行訓練。其訓練目的在於獲得訓練一個精確的分類模型，使得當給予未知查詢時，能夠正確地給予文件集中所有的文件最適當的分類(及相關或不相關)。待分類完成之後，可根據標籤為相關者排序於標籤為不相關者之前之原則，進行排序，這個分類模型，即為所謂的排序模型。

不論是使用機器學習方法之訓練模型或是傳統的檢索模型，大多使用平均精確率(Mean Average Precision, MAP)與均化遞減累積獲益(Normalized Discount Cumulative Gain, NDCG)做為評估工具。在對「查詢/文件對組」中所有擷取的特徵進行說明之前，我們先對實驗的評估工具及語料庫進行介紹，因為所擷取的特徵中部份為傳統的資訊檢索方法。在介紹傳統資訊檢索方法時，我們也同時呈現其檢索效能。

3.2 評估工具

■ 平均精確率(Mean Average Precision, MAP)

平均精確率一直是資訊檢索重要的評估方法之一。平均精確率目標是為了評估檢索系統所回傳文件在相關性上的精確程度。不僅衡量檢索系統是否能搜尋出相關文件，同時也衡量被搜尋出的相關文件之排名是否在前面，若相關文件之排序越

前面，則表示此排名情況越佳。所以，相關文件位於檢索系統回傳文件序列上的排名順序將會影響到平均精確率的值。平均精確率越高的結果，代表檢索系統能找出越多相關的文件，並且這些相關文件在回傳文件序列的排名越前面。在說明平均精確率之前，我們先行了解精確率之算法，給定一個查詢時，精確率($p@n$)算法如下:

$$P@n = \frac{\text{在檢索出的前}n\text{篇文件之中實際為相關之文件數}}{n} \tag{3.2.1}$$

舉例來說，如果給予一個查詢，透過排序函數，前十篇文件之結果如下：$\{d_2, d_{10}, d_5, d_3, d_4, d_1, d_7, d_8, d_6, d_9\}$，而其正確相關文件為$\{d_2, d_3, d_6, d_{10}\}$，則排序結果與相關對應為：

{相關,相關,不相關,相關,不相關,不相關,不相關,不相關,相關,不相關} ，因此 $P@1$ 至 $P@10$ 之值為 $\{1, 1, 2/3, 3/4, 3/5, 3/6, 3/7, 3/8, 4/9, 4/10\}$。

對於單一的查詢，對 $P@n$ 做平均則可以得到平均精確率：

$$AvgP_i = \frac{\sum_{n=1}(P@n) \times rel(n)}{\text{所有與}q_i\text{為相關之文件數}}$$

$$rel(n) = \begin{cases} 1 & \text{若第}n\text{篇文件為相關} \\ 0 & otherwise \end{cases} \tag{3.2.2}$$

以上例而言， $AvgP = \frac{1+1+3/4+4/9}{4} = 0.7986$ 。對於所有 k 個的查詢之平均精確率可表示成：

$$\text{MAP} = \frac{\sum_{i=1}^{k} AvgP_i}{k} \tag{3.2.3}$$

■ 均化遞減累積獲益(Normalized Discount Cumulative Gain, NDCG)

$P@n$ 與 MAP 只能針對「相關」、「不相關」兩種狀態做評估。而近幾年提出之

均化遞減累積獲益(NDCG)[Järvelin & Kekäläinen 2002]則可以處理有多種層次之相關的問題評估。因此，均化遞減累積獲益不僅可以使用在只有「相關」與「不相關」兩種狀態作評估，亦可對有更多「相關」與「不相關」分級狀態之問題作評估。其計算方式如下：

$$N@n = Z_n \sum_{j=1}^{n} \begin{cases} 2^{r(j)} - 1, & j = 1,2 \\ \dfrac{2^{r(j)} - 1}{log(j+1)}, & j > 2 \end{cases} \qquad (3.2.4)$$

j 為排名位置 j 之文件，$r(j)$ 為此文件的實際分數，Z_n 為均化項。其中，$r(j)$ 之分數越相關分數越高，例如：「相關」分數給定為 1：「不相關」分數給定為 0。而 Z_n 則確保最佳的排序評估結果為 1，為一均化項。此方法納入相關文件的排名位置之重要性概念，亦即，如果與查詢高度相關的文件排名在後面，即使其分數 $r(j)$ 很高，仍然會因為其排序位置不佳，而降低分數。我們以只有「相關」、「不相關」兩種狀態下舉例。當對於一個查詢排序結果之前 10 篇文件 $\{d_2, d_{10}, d_5, d_3, d_4, d_1, d_7, d_8, d_6, d_9\}$。若實際上為相關文件，其 $r(j) = 1$，反之則 $r(j) = 0$。這 10 篇文件中之 $\{d_2, d_3, d_6, d_{10}\}$ 為相關。因此，對這 10 篇文件之分數為 $\{1,1,0,1,0,0,0,0,1,0\}$。各別位置之累積獲益為 $\{1,1,0,\frac{1}{\log(4)},0,0,0,0,\frac{1}{\log(9)},0\}$。對此一查詢於位置 10 之均化遞減累積 $N@10 = Z_n\left(1+1+\frac{1}{log(4)}+\frac{1}{log(9)}\right)$。對所有的查詢之前 10 篇文件皆做此計算之後取平均，即可以得到所有查詢之 $N@10$ 資訊。

■ 平均精確率與均化遞減累積獲益之討論

在本論文中，我們與 Nallapati 於一個查詢對應文件分為「相關」與「不相關」之概念相同[Nallapati 2004]。因此，在沒有階層性相關度的情況下，我們必須探討平均精確率與均化遞減累積獲益之關係，若平均精確率與均化遞減累積獲益為

正相關，則進行兩種評估並無太大的意義。然則，我們發現平均精確率和均化遞減累積獲益並非是正相關。為驗證此關係，我們需要了解，在兩種排列序列中，相同的位置觀察下，是否有平均精確率提升時，均化遞減累積獲益卻減低的情況。我們試舉一例來說明，假設文件集中共有十篇文件，其中兩篇跟某一查詢相關，其餘八篇則不相關。如今有兩個排序演算法(A、B)傳回各自的序列，其中 A 產生的序列中，兩篇相關文件分別被排在第二名及第九名；B 產生的序列中，兩篇相關文件分別被排在第三名及第四名，如下所示：

A: 0 1 0 0 0 0 0 0 1 0

B: 0 0 1 1 0 0 0 0 0 0

試問，何者有較佳的檢索成效？ 若我們分別對兩序列計算精確度及遞減累積獲益，可得表 3.2，其中 $Z_{10} = \frac{1}{2}$：

	MAP	NDCG@10
A	(1/2+2/9)/2=0.361	$(\frac{1}{2}) \cdot \left(1 + 1/\log 10\right) = 0.6505$
B	(1/3+2/4)/2=0.417	$(\frac{1}{2}) \cdot \left(1/\log 4 + 1/\log 5\right) = 0.4653$

表 3.2 不同序列結果之平均精確率與遞減累積獲益

B 有較高的精確度，但 A 有較高的遞減累積獲益。由此可知，當精確率增加時，遞減累積獲益並不一定會同時增加，所以，平均精確率增加時，均化遞減累積獲益未必呈現正成長。因此平均精確率與均化遞減累積獲益可以補足觀察到不同的現象。

3.3 實驗語料

本論文使用了兩套 TDT 語料[LDC 2000]，分別為 TDT-2 和 TDT-3，TDT (Topic

Detection and Tracking)為美國國防部先進研究計劃機構 (Defense Advanced Research Projects Agency, DARPA)所資助的研究。關於，TDT-2 及 TDT-3 語料的各種統計資料為表 3.3。

TDT-2 中包含有新華社新聞(XIN)、美國之音中文廣播新聞(VOA)以及聯合中文網(ZBN)，在本論文中取用新華社新聞及美國之音中文廣播新聞，並使用 TDT-2 中美國之音中文廣播新聞的語音辨識結果。美國之音中文廣播新聞以及新華社新聞的擷取時間點由 1998 年 1 月至 6 月之間，其中，美國之音中文廣播新聞 1 月 1 日至 2 月 19 日並無新聞文件資料。TDT-2 中定義有 20 個主題(Topics)，每個主題都是一篇新華社新聞，我們將這 20 個主題作為 20 個測試查詢。這 20 個主題的相關文件分布在新華社新聞、美國之音廣播新聞及聯合中文網。然而，美國之音中文廣播新聞才有語音文件，而部份主題並沒有和美國之音中文廣播新聞相關。因此，我們刪除了與美國之音廣播新聞無關的主題，擷錄 16 個主題，作為實驗中所使用的測試查詢。而訓練查詢，則選用新華社新聞共 813 則新聞。

	TDT-2 1998, 02~06			TDT-3 1998, 10~12		
語音文件	2,265 則, 46.03 小時之語音檔			3371 則, 98.43 小時之語音檔		
查詢	測試查詢		訓練查詢	測試查詢		訓練查詢
	16 則 新華社新聞		831 則 新華社新聞	47 則 新華社新聞		777 則 新華社新聞
	最少	最多	平均	最少	最多	平均
語音文件長度(字)	23	4841	287.1	19	3667	415.1
測試查詢長度(字)	183	2623	532.9	98	1477	443.6
與測試查詢相關之文件數	2	95	29.3	3	89	20.1

表 3.3 實驗語料資訊

同樣的，亦使用 TDT-3 中的新華社新聞以及美國之音中文廣播新聞，搜集語料庫的時間點為 1998 年 10 月至 12 月之間。在 TDT-3 中定義有 60 個主題亦即 60 個測試查詢，每一個主題皆為一則新華社新聞。我們同樣選刪除了與美國之音廣播新聞無關的主題，擷錄 47 個主題作為實驗中所使用的測試查詢。訓練查詢亦選用新華社新聞共 777 則新聞。

3.4 特徵選取

特徵選取中，我們選用了共 42 組特徵，共分為：低階特徵(Low-level Features)、相近度特徵(Proximity Features)以及機率特徵(Probabilistic Features)三類。以下我們將分別進行介紹。

3.4.1 低階特徵(Low-level Features)

本論文使用之低階特徵是參考自[Liu et al. 2007]中所被使用之 10 組低階特徵，其結合 Nallapati 提出之 6 種特徵[Nallapati 2004]以及[Baeza-Yates & Ribeiro-Neto 1999]中介紹之特徵。除此 10 組特徵之外，本論文額將文件長度特徵，納入為低階特徵。

■ LF1：

查詢在文件出現的頻率，亦稱為詞頻。通常我們認為文件中若出現與查詢相同的文字，且出現次數越高則越有可能與此查詢相關。詞頻為一種簡單卻顯明之特徵，因此我們選用詞頻為一特徵，其數學表示為式(3.4.1)。

$$\mathrm{LF1} = \sum_{q_i \in Q \cap D} c(q_i, D) \qquad (3.4.1)$$

其中，Q 為查詢的詞集合，D 為文件的詞集合，q_i 為查詢與文件皆有出現的詞。$c(q_i, D)$ 為 q_i 出現在 D 的次數。

■ LF2：

LF2 對詞頻標準化(Normalization)。查詢中的詞出現於文件中的次數越高固然越好，但是也可能因為文件本身的長度就已經較長，而導致次數也相對較多。某查詢詞 q_i 在兩篇文件 D_j 及 D_l 中皆出現同樣次數的詞頻，但是 D_j 的長度 $|D_j|$ 大於 D_l 的長度 $|D_l|$，則詞頻標準化的情形下，會認為 q_i 在 D_j 的重要性大於 D_l。其數學表示式為式(3.4.2)。

$$\text{LF2} = \sum_{q_i \in Q \cap D} \frac{c(q_i, D)}{|D|} \tag{3.4.2}$$

其中，Q 為查詢的詞集合，D 為文件的詞集合，q_i 為查詢與文件皆有出現的詞。$c(q_i, D)$ 為 q_i 出現在 D 的次數，$|D|$ 為文件長度。

■ LF3：

LF3 為反文件詞頻(Inverse Document Frequency)。如果一個詞僅在少部份文章中才出現，其重要性比其它在每一篇文章中皆出現的詞來得重要。例如：「的」幾乎在每一篇文章中都出現，這樣的詞無法代表不同文件的重要特性：相反的，「總統」這樣的詞在部份文件，例如政治類文件才會出現，而講述綜藝類新聞的文件不容易出現此詞，所以，這是一種鑑別能力較高的詞。因此，我們可以將反文件詞頻視為一個特徵。其數學表示式如式(3.4.3)。

$$\text{LF3} = \sum_{q_i \in Q \cap D} idf(q_i) \qquad idf(q_i) = \log \frac{N}{n(q_i)} \tag{3.4.3}$$

其中，Q 為查詢的詞集合，D 為文件的詞集合，q_i 為查詢與文件皆有出現的詞，N 為語料庫中的文章總數，n 為有出現 q_i 的文章篇數。$idf(q_i)$ 分數高時，代表詞 q_i 鑑別力較高：反之，則鑑別力較低。

■ LF4：

如果詞在整體語料庫中佔的比例過大，則此詞極有可能在各種類別的文章中皆會出現，因此，將詞在整體語料庫中之比例作倒數的轉換，這樣就可以展現出詞的稀有性質。其數學表示式如式(3.4.4)。

$$\text{LF4} = \sum_{q_i \in Q \cap D} \log\left(\frac{|C|}{c(q_i, C)}\right) \tag{3.4.4}$$

其中，Q 為查詢的詞集合，D 為文件的詞集合，C 為整體語料庫，$|C|$ 為整體語料庫的總字數，q_i 為查詢與文件皆有出現的詞，$c(q_i, C)$ 為 q_i 出現在整體語料庫的次數。此可以看出某一詞彙在整體語料庫的文字數中所占的比例。在整體語料庫中出現次數越高時，此特徵值越小，代表詞的稀有性質較低；反之，特徵值越大，代表詞的稀有性質較高。

■ LF5：

LF5 將 LF1 特徵值的範圍區間做對數調整。其數學表示式為式(3.4.5)。

$$\text{LF5} = \sum_{q_i \in Q \cap D} log\left(c(q_i, D)\right) \tag{3.4.5}$$

其中，Q 為查詢的詞集合，D 為文件的詞集合，q_i 為查詢與文件皆有出現的詞。$c(q_i, D)$ 為 q_i 出現在 D 的次數。

■ LF6：

同時考慮 LF2 及 LF5，當某一詞在一篇文章中之分布佔整篇文章之比例很高，並且它在整體語料庫中其實並不常出現時，這樣代表它和此篇文件的相關度極高，可用以作為區別此篇文件與其它文件。其數學表示式為式(3.4.6)。

$$\text{LF6} = \sum_{q_i \in Q \cap D} log\left(1 + \frac{c(q_i, D)}{|D|} \frac{|C|}{c(q_i, C)}\right) \tag{3.4.6}$$

其中，Q 為查詢的詞集合，D 為文件的詞集合，C 為整體語料庫，$|D|$ 為文件 D 之長度，$|C|$ 為整體語料庫的總字數，q_i 為查詢與文件皆有出現的詞，$c(q_i, D)$ 為 q_i 出現在 D 的次數，$c(q_i, C)$ 為 q_i 出現在整體語料庫的次數。

■ LF7：

LF7 將 LF2 特徵值的範圍區間做對數調整。其數學表示式為式(3.4.7)。

$$\text{LF7} = \sum_{q_i \in Q \cap D} \log(1 + \frac{c(q_i, D)}{|D|}) \qquad (3.4.7)$$

其中，Q 為查詢的詞集合，D 為文件的詞集合，$|D|$ 為文件 D 之長度，$c(q_i, D)$ 為 q_i 出現在 D 的次數。

■ LF8：

同時考慮 LF2 和 LF3。LF8 考慮 q_i 出現於多少文章之篇數，如果出現的篇數過多，則視為此 q_i 極為普遍，其 $idf(q_i)$ 較小。其數學表示式為式(3.4.8)。

$$\text{LF8} = \sum_{q_i \in Q \cap D} \log(1 + \frac{c(q_i, D)}{|D|} idf(q_i)) \qquad idf(q_i) = \log \frac{N}{n(q_i)} \qquad (3.4.8)$$

其中，D 為文件的詞集合，$|D|$ 為文件 D 之長度，q_i 為查詢與文件皆有出現的詞，$c(q_i, D)$ 為 q_i 出現在 D 的次數，N 為語料庫中的文章總數，n 為有出現 q_i 的文章篇數。

■ LF9：

LF9 將 LF3 特徵值的範圍區間做對數調整。其數學表示式為式(3.4.9)。

$$\text{LF9} = \sum_{q_i \in Q \cap D} log(idf(q_i)) \qquad idf(q_i) = log \frac{N}{n(q_i)} \qquad (3.4.9)$$

Q 為查詢的詞集合，D 為文件的詞集合，q_i 為查詢與文件皆有出現的詞，N 為

語料庫中的文章總數，n 為有出現 q_i 的文章篇數。

■ LF10：

結合 LF3 及 LF5，其數學表示式為式(3.4.10)。

$$\text{LF10} = \sum_{q_i \in Q \cap D} c(q_i, D)\, idf(q_i) \qquad idf(q_i) = \log \frac{N}{n(q_i)} \tag{3.4.10}$$

Q 為查詢的詞集合，D 為文件的詞集合，$c(q_i, D)$ 為 q_i 出現在 D 的次數，N 為語料庫中的文章總數，n 為有出現 q_i 的文章篇數。

■ LF11：

當文件的長度越長時，通常會包含更多的資訊，文件的長度越短時，則包含的資訊較少。在[Singhal et al. 1996]的研究中，說明索文件長度的資訊影響資訊檢索的成效甚大，因此我們考慮文件的長度做為其中之一的特徵。其數學表示式為式(3.4.11)。其中，$|D|$ 為文件 D 之長度。

$$\text{LF11} = |D| \tag{3.4.11}$$

3.4.2 相近度特徵(Proximity Features)

■ Proximity-VSM：

向量空間模型法[Baeza-Yates & Ribeiro-Neto 1999]在 1968 年由 Salton 提出，最先使用在 SMART(System for the Mechanical Analysis and Retrieval of Text)資訊檢索系統上，向量空間模型法的目的在於測量查詢和文件之間相似程度，據此給予查詢和文件一個相關分數，並利用此分數進行文件的排名。向量空間模型法的做法是將文件和查詢都用向量表示，估測文件向量和查詢向量的餘弦分數，分數較高者代表為較為相關的文件。對於一篇文件的向量可以以下列方式表示:

$$\vec{d}_j = (w_{1,j}, w_{1,j}, ..., w_{t,j}) \tag{3.4.12}$$

對於查詢的向量可以以下列方式表示:

$$\vec{q} = (w_{1,q}, w_{2,q}, ..., w_{t,q}) \tag{3.4.13}$$

其中 t 代表所有在資訊檢索系統中的詞典的詞彙總數，$w_{i,j}$ 代表權重值，通常 $w_{i,j}$ 的計算方式為:

$$w_{i,j} = tf_{i,j} \times idf_i \tag{3.4.14}$$

$$tf_{i,j} = \frac{freq_{i,j}}{\max_l freq_{l,j}} \tag{3.4.15}$$

$$idf_i = \log \frac{N}{n_i} \tag{3.4.16}$$

其中 $\max_l freq_{l,j}$ 代表在文件 d_j 中出現頻率最高的詞 l 的頻率，而 N 代表所有文件的數目，n_i 則是詞 i 出現在多少篇文件中。$tf_{i,j}$ 為詞頻(Term Frequency)，如果詞 i 在文件 d_j 中出現次數很高，則 $tf_{i,j}$ 越高；idf_i 為反文件頻率(Inverse Document Frequency)，反應了詞 i 的鑑別資訊能力，如果詞 i 在大部份的文件中皆有出現，很有可能是一種虛詞，例如:的，因此，idf_i 值越高，代表詞 i 僅特別出現在某部份的文件之中，因此，其鑑別力高，此資訊較為重要。

建立了文件以及查詢的向量，就可以利用餘弦(Cosine)定理估量文件及查詢的相關程度，其式如下:

$$\text{VSM-0:} \quad sim(d_j, q) = \frac{\vec{d}_j \bullet \vec{q}}{|\vec{d}_j| \times |\vec{q}|} = \frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2} \times \sqrt{\sum_{i=1}^{t} w_{i,q}^2}} \tag{3.4.17}$$

其中 $|\vec{d}_j|$ 及 $|\vec{q}|$ 分別代表文件的向量長度以及查詢的向量長度。圖 3.1 為向量空間模型之示意圖。

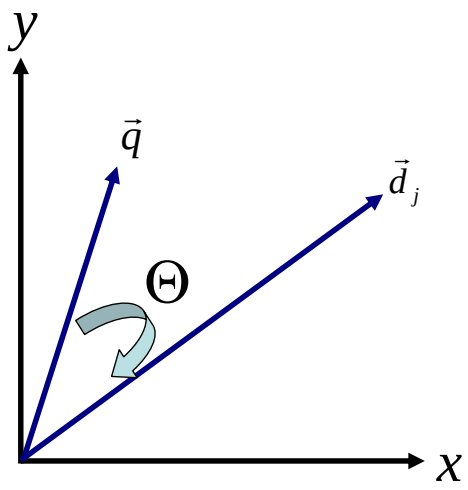

圖 3.1 向量空間模型之物理示意圖

$\vec{d}_j$及$\vec{q}$在向量空間中之夾角Θ越小，其計算的$sim(d_j, q)$的分數越高；反之，Θ越大，計算的$sim(d_j, q)$的分數越低。各別完成在文件集D中之所有文件與查詢q之分數估量之後，以往的向量空間模型，即可以此分數做為排序，其排序結果即是檢索結果。

除了式(3.4.17)，本論文也試著向量空間模型做不同的權重的延伸，其調整的數學式如下：

VSM-1: $$sim(d_j, q) = \frac{\vec{d}_j \bullet \vec{q}}{|\vec{d}_j| \times |\vec{q}|} = \frac{\sum_{i=1}^{t} (w_{i,j})^2 \times (w_{i,q})^2}{\sqrt{\sum_{i=1}^{t} (w_{i,j}^2)^2} \times \sqrt{\sum_{i=1}^{t} (w_{i,q}^2)^2}} \quad (3.4.18)$$

VSM-2: . $$sim(d_j, q) = \frac{\vec{d}_j \bullet \vec{q}}{|\vec{d}_j| \times |\vec{q}|} = \frac{\sum_{i=1}^{t} (w_{i,j})^{1/2} \times (w_{i,q})^{1/2}}{\sqrt{\sum_{i=1}^{t} (w_{i,j}^{1/2})^2} \times \sqrt{\sum_{i=1}^{t} (w_{i,q}^{1/2})^2}} \quad (3.4.19)$$

式(3.4.18)及式(3.4.19)是對權重值$w_{i,j}$進行調整，目的僅是為了在訓練時能夠獲得更多可見資訊。

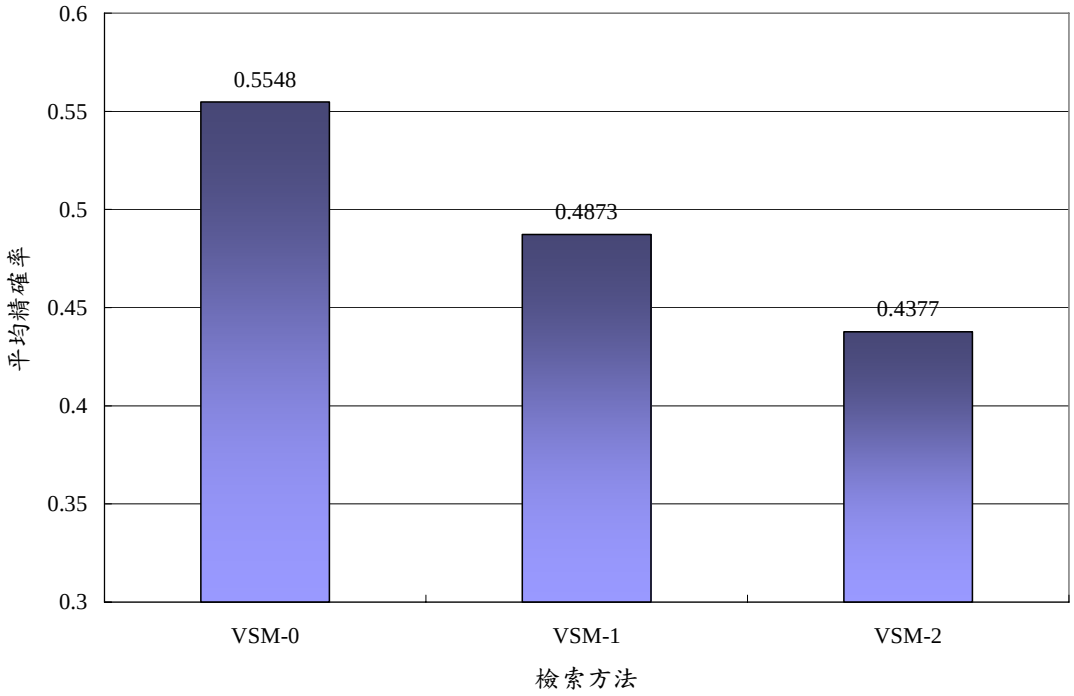

圖 3.2 實驗於 TDT-2 語音正確轉寫文件 VSM 之 MAP

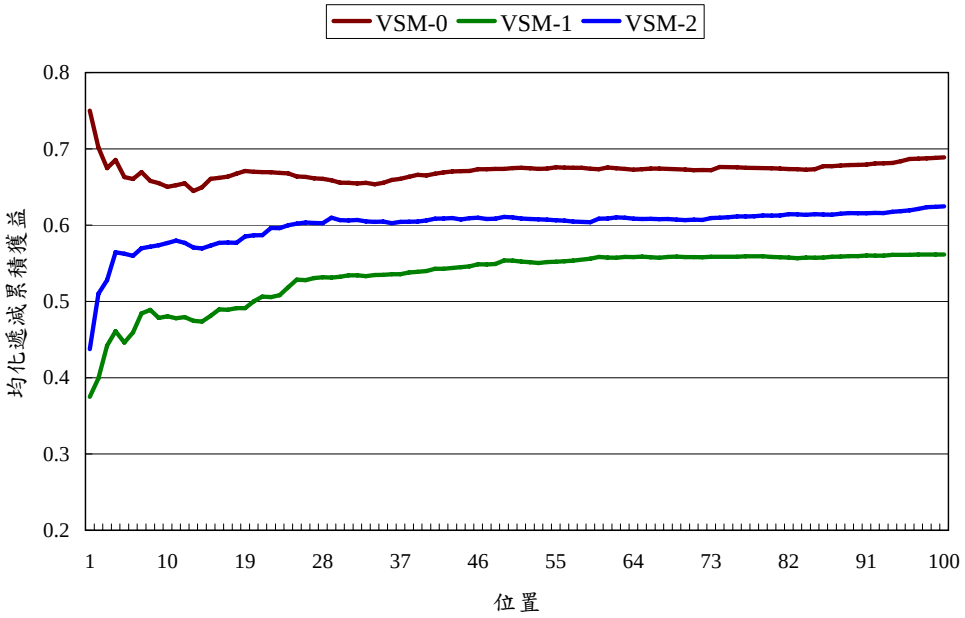

圖 3.3 實驗於 TDT-2 語音正確轉寫文件 VSM 之 NDCG 曲線

當使用於支援向量機進行文件檢索時，我們將向量空間模型的相似分數當做

一個特徵值，而經由不同權重的延伸，期望在進行訓練時可以看到更多的資訊。接著，我們展示單獨以向量空間模型進行文件檢索所得出之平均精確率，如圖 3.2。圖 3.3 中，其為在不同位置下的均化遞減累積獲益。雖然 VSM-1 及 VSM-2 明顯不及 VSM-0，但我們期望此二者能有不同層面的資訊是傳統 VSM 所沒有的，最後即使證明此二特徵沒有包含有用的資訊，我們相信在機器學習的過程中也會進行自動篩選(意指減弱其權重)，不會對訓練的模型產生傷害。

■ Proximity-LSI：

在從事資訊檢索時，經常遇到查詢與文件在詞面上不相同，但是實際上卻是隱含著相同意義的情況。例如：「美國」以及「紐約」在詞面上完全不同，但是這兩個詞彙實質上是有一定的相關程度，而隱藏語意索引(Latent Semantic Indexing, LSI) [Furnas et al. 1988]針對這樣的問題進行改進，期望在字面吻合之外尋求意義上相關的資訊。因此，隱藏語意索引最主要的概念在於將每一則文件向量及查詢向量都轉置到較低維度的空間中，而此較低維度空間是含有語意資訊的。如此一來，就能夠運用到隱藏語意的資訊。

隱藏語意索引的作法，首先，先將文件群轉換成為一個「詞×文件」之矩陣，矩陣中的值可以是 $tf \times idf$ ，這樣的矩陣經常為一個稀疏矩陣(sparse matrix)，接著對此矩陣進行奇異值分解 (Singular Value Decomposition, SVD)，如圖 3.4。而矩陣經過奇異質分解之數學表示式為式(3.4.20)，其中 X 為「詞×文件」，為一 $M \times N$ 之矩陣，m 代表詞總數，n 代表文件總數。經過分解後，T_r 及 D_r^t 皆為固有向量(Eigen-vector)所組成之矩陣，S_r 為固有值(Eigen-value)組成之對角化(Diagonal)矩陣。T_r 為 $M \times r$ 之矩陣，S_r 為 $r \times r$ 矩陣，D_r^t 為 $r \times N$ 矩陣。其中，r 為隱藏主題之意含。

$$X = T_r S_r D_r^t \tag{3.4.20}$$

當「詞×文件」矩陣可以經由奇異質分解後，我們就可以從分解後的矩陣中，

選擇最重要的主題資訊。因此，先決定一個維度k，擷取前k維用以接近原本的稀疏矩陣，如式(3.4.21)。此k設定為 1 時，表示使用 1 個主題，並且此主題相較於其它主題為最主要的資訊，但其它剩餘的資訊就會被忽略，因此遺失部份資訊。

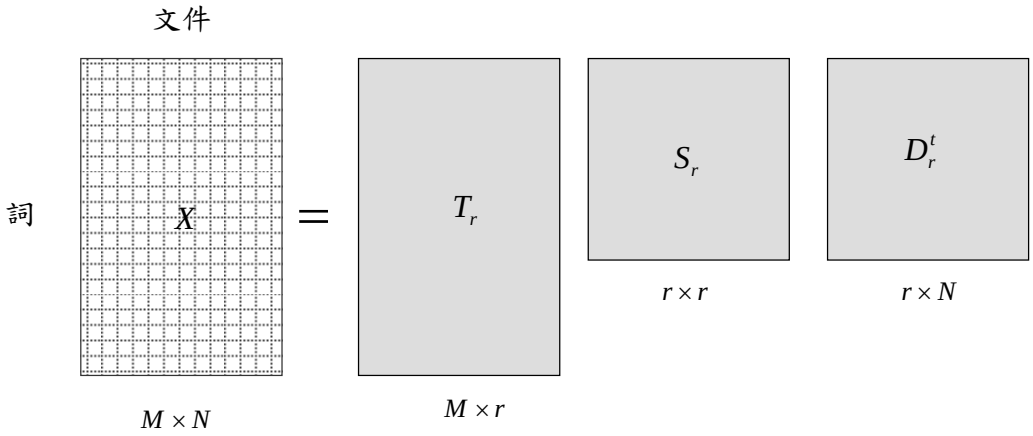

圖 3.4 矩陣奇異值分解示意圖

當k設定越來越大時，使用的主題越來越多，直到k設定最大至r時，表示使用了全部的主題資訊，沒有資料遺失，但同時也喪失了利用奇異質分解找出較重要主題資訊的意義。因此，維度k的選擇，可以幫助我們運用適當的主題資訊，有助於提高檢索的效能。隱藏語意索引的實驗中，對k的選擇不同，會得到不同的檢索結果。圖 3.5 為選擇k後的降維示意圖。

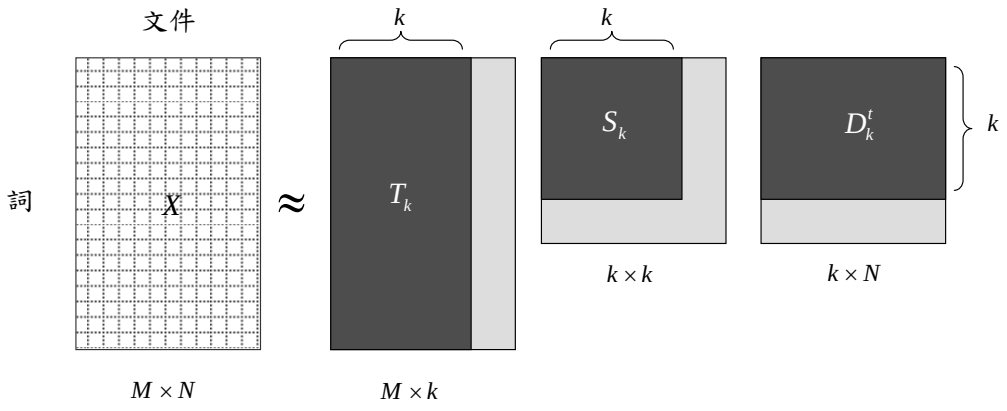

圖 3.5 矩陣奇異值分解並降維示意圖

$$\hat{X} = T_k S_k D_k^t \approx X \tag{3.4.21}$$

在選定好維度k後，同時也決定了T_k、S_k及D_k^t。接著，我們利用T_k、S_k及D_k^t建構文件向量$\vec{d}$及查詢向量$\vec{q}$。對文件向量$\vec{d}$，我們可以使用D_k^t的資訊。由於D_k^t中對每一篇文件皆有一組k維的主題向量，因此我們可以選用此向量作為文件向量$\vec{d}$。對於查詢向量$\vec{q}$，我們可以先建立一組向量如(3.4.13)，再對此向量進行轉換，其轉換方式為式(3.4.22)。因此，我們就可以得到含有隱藏語意的查詢。處理好

$$\vec{q}'_{1\times k} = (\vec{q})_{1\times m} T_{m\times k} \tag{3.4.22}$$

文件向量及查詢向量後，便可以使用餘弦評估方式，估算文件及查詢的相關度，其數學表示式為式(3.4.23)。餘弦評估後分數越高者，則代表越相關。

$$sim(d,q) = \frac{\vec{d} \bullet \vec{q}'}{\left|\vec{d}\right| \times \left|\vec{q}'\right|} \tag{3.4.23}$$

當使用於支援向量機進行訓練時，我們將隱藏語意索引的分數當做一種特徵值。由於隱藏語意索引可以設定不同的維度，接著，我們展示單獨以隱藏語意索引，

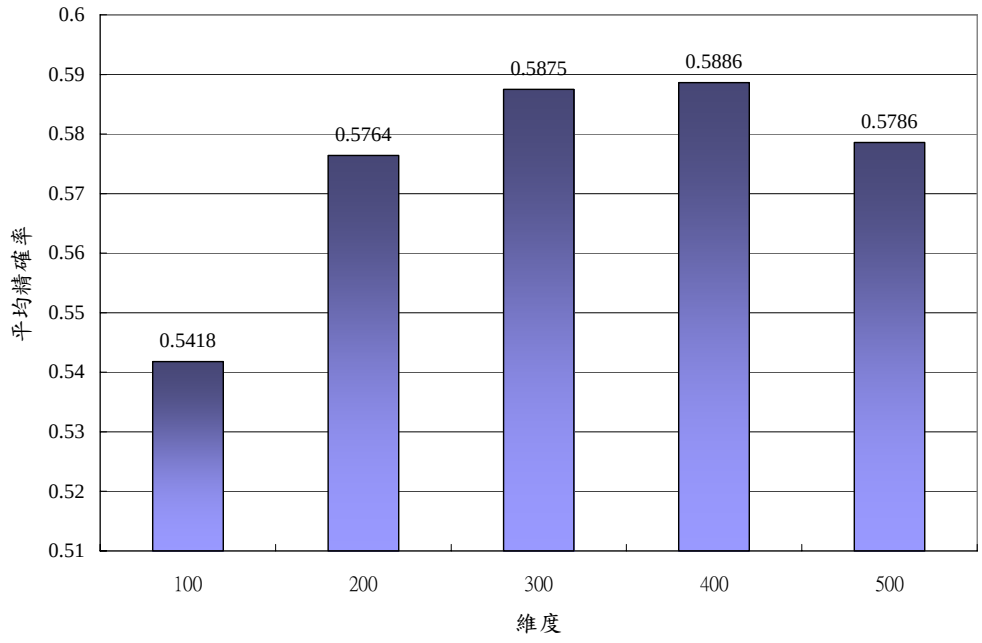

圖 3.6 實驗於 TDT-2 語音正確轉寫文件 LSI 各維度之 MAP

在 TDT-2 語料庫中，進行文件檢索所得出之平均精確率，如圖 3.6。其均化遞減累積獲益曲線如圖 3.7 所示。從圖 3.6 中我們可以得知，選取過多的維度，不一定能夠提高平均精確率；而太少的維度喪失太多資訊，也會造成平均精確率過低。而在圖 3.7 中，我們也可以得知，維度給定為 300 時，其均化遞減累積獲益效果較維度 100 及維度 200 為好。

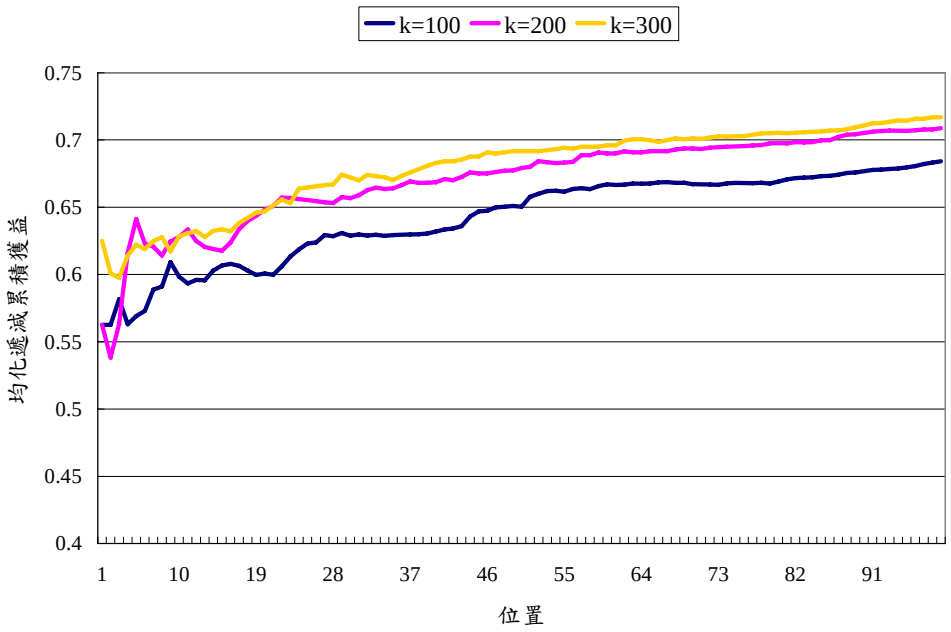

圖 3.7 實驗於 TDT-2 語音正確轉寫文件 LSI 各維度之 NDCG

3.4.3 機率模型(Probabilistic Features)

■ Probabilistic-BM25：

最佳配對權重最先由 Roberson 與 Sparck Jones 所提出，由最剛開始的 BM1，延伸至 BM15 及 BM11，最後結合了 BM15 及 BM11 形成單一的函數，即為 BM25[Roberson et al. 1995]，而 1998 年 Beaulieu 和 Jones 將其應用在 Okapi 資訊檢索系統之中，而形成了目前的 Okapi BM25，其數學表示式為式(3.4.24)。其中 q_i 為查詢 q 中的某一個詞，$n(q_i)$ 為所有文件 D 中包含 q_i 詞的文件數，$|d_j|$ 則為文

件 d_j 之文件長度，*avrgl* 為所有文件 D 的平均文件長度，$f(q_i, d_j)$ 為 q_i 出現在文

$$Score(d_j, q) = \sum_{i=1}^{n} IDF(q_i) \cdot \frac{f(q_i, d_j) \cdot (k+1)}{f(q_i, d_j) + k \cdot (1 - b + b \cdot \frac{|d_j|}{avrgl})} \qquad (3.4.24)$$

$$IDF(q_i) = \log \frac{N}{n(q_i)} \qquad (3.4.25)$$

件 d_j 的次數，k 及 b 為調整之參數，而式(3.4.24)中之 $IDF(q_i)$ 數學表示式為(3.4.25)。BM25 模型，不僅運用了詞頻的分數，更考慮文件的長度，並且對文件長度進行標準化的動作,了解此篇文件在所有的文件中是屬於較長的文件亦或是較短的文件。

BM25 中可以調整 k 及 b 兩項參數，表 3.4 為在 TDT-2 語料庫中，使用不同的參數設定下的平均精確率實驗結果。圖 3.8 為在 TDT-2 語料庫中，使用不同的參數設定下的均化遞減累積獲益結果。我們可以發現，在實驗中，當參數值 k 設定為 0.1 而 b 設定為 0.01 時，可以得到最好的平均精確率及均化遞減累積獲益結果。然而，一般 BM25 的設定為 k =2；b =0.75，這樣的設定反而在我們的語料庫

模型名稱	*k*	*b*	MAP
BM25_01_001	0.1	0.01	0.5846
BM25_05_001	0.5	0.01	0.5521
BM25_10_001	1.0	0.01	0.5156
BM25_01_050	0.1	0.50	0.5559
BM25_01_005	0.1	0.05	0.5856
BM25_01_010	0.1	0.10	0.5857
BM25_20_075	2.0	0.75	0.3515

表 3.4 實驗於 TDT-2 語音正確轉寫文件 BM25 各種參數設定的 MAP

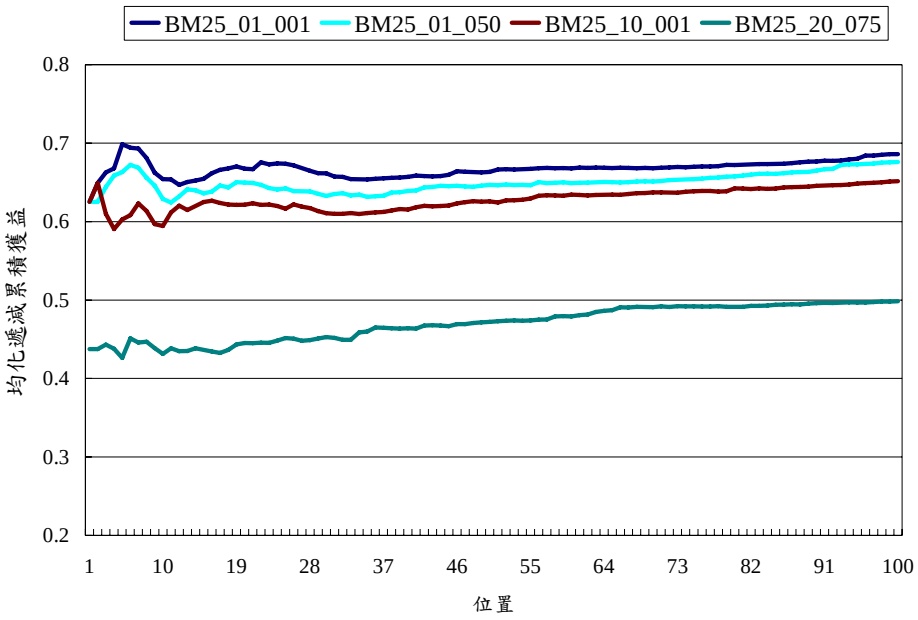

圖 3.8 實驗於 TDT-2 語音正確轉寫文件 BM25 各種參數調整下之 NDCG

中成效最差。由此也可以突顯出傳統資訊檢索的問題：在不同語料上，其實是很難決定如何設定參數，必須要經過不斷的調整才可能得知。

當使用於支援向量機進行訓練時，我們將 BM25 各項參數設定結果的分數納為多個特徵。以此方法，將多種參數設定結果一同進行訓練。

■ Probabilistic-LM：

語言模型(Language Model, LM)運用在資訊檢索上，其主要的概念是假設一個查詢 q 經由機率模型，由一篇文件 d 所生成[Zhai & Lafferty 2001]。給定一個查詢 q，跟一則文件 d，我們嘗試著去估計其條件機率 $p(d|q)$。有了條件機率 $p(d|q)$，我們使用貝式定理轉換之後，可以得到下列式子：

$$p(d|q) \propto p(q|d)p(d) \qquad (3.4.26)$$

其中，$p(d)$ 為事前機率，我們經常視為一致(Uniform)，亦即每一則文件的 $p(d)$ 皆相同，因此，實質上我們僅需計量 $p(q|d)$ 之機率。在[Zhai & Lafferty 2001]中，對 $p(q|d)$ 提出了平滑化(Smoothing)的方法：Jelinek-Mercer LM，其數學式為：

$$p(q|d) = (1-\lambda)p_{ml}(q_i|d) + \lambda p(q_i|C) \quad (3.4.27)$$

其中，λ 為一可調整之參數，其值介於 0~1 之間；q_i 為查詢中的詞；C 為整體語料庫；$p_{ml}(q_i|d)$ 為文件 d 產生 q_i 之機率，可以使用 q_i 在文件 d 出現次數與文件 d 總體字數之比例作為此機率值；$p(q_i|C)$ 為整體語料庫產生 q_i 的機率。以下，我們試著調整不同的 λ 值，實驗於 TDT-2 語料中，觀察其檢索效果。表 3.5 為平均精確率之檢索效能。

模型名稱	λ	MAP	模型名稱	λ	MAP
LM001	0.01	0.5517	LM007	0.07	0.6108
LM002	0.02	0.5872	LM008	0.08	0.6106
LM003	0.03	0.6026	LM009	0.09	0.6113
LM004	0.04	0.6099	LM010	0.10	0.6133
LM005	0.05	0.6110	LM020	0.20	0.5940
LM006	0.06	0.6104	LM090	0.90	0.4421

表 3.5 實驗於 TDT-2 語音正確轉寫文件 LM 在各種參數調整下之 MAP

圖 3.9 為均化遞減累積獲益之結果。從結果中，我們可以得知，當 λ 設定為 0.1 時，能夠達到最佳的檢索效能。而當 λ 設定<0.1 時，平均精確率降低；當設定>0.1 時，平均精確率更大幅度降低，這表示 $p(q_i|C)$ 影響較大，但也不能完全的依賴 $p(q_i|C)$，$p_{ml}(q_i|d)$ 仍然有貢獻資訊。

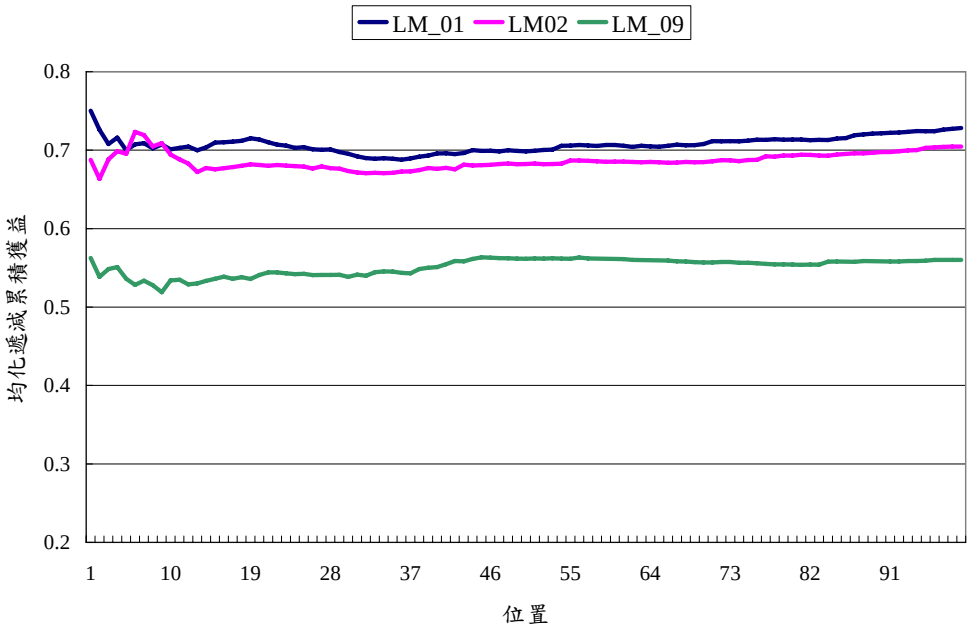

圖 3.9 實驗於 TDT-2 語音正確轉寫文件 LM 在各種參數調整下之 NDCG

總結本節的所有特徵，我們總共選取了 42 個特徵，特徵之選定及各項特徵的參數設定如表 3.6 所示。其中，編號 18 至編號 24 特徵，分別為編號 11 至邊號 17，使用 BM25 作為特徵的特徵值範圍做對數調整。

特徵編號	特徵名稱	特徵說明
1	LF1	
2	LF2	
3	LF3	
4	LF4	
5	LF5	
6	LF6	
7	LF7	
8	LF8	
9	LF9	
10	LF10	
11	BM25_01_001	參數設定：b=0.1；k=0.01
12	BM25_01_010	參數設定：b=0.1；k=0.1
13	BM25_10_001	參數設定：b=1；k=0.01

14	BM25_01_005	參數設定：b=0.1；k=0.05
15	BM25_01_050	參數設定：b=0.1；k=0.5
16	BM25_05_001	參數設定：b=0.5；k=0.01
17	BM25_20_075	參數設定：b=2；k=0.75
18	logBM25_01_001	參數設定：b=0.1；k=0.01
19	logBM25_01_010	參數設定：b=0.1；k=0.1
20	logBM25_10_001	參數設定：b=1；k=0.01
21	logBM25_01_005	參數設定：b=0.1；k=0.05
22	logBM25_01_050	參數設定：b=0.1；k=0.5
23	logBM25_05_001	參數設定：b=0.5；k=0.01
24	logBM25_20_075	參數設定：b=2；k=0.75
25	LF10	
26	LM001	參數設定：λ=0.01
27	LM002	參數設定：λ=0.02
28	LM003	參數設定：λ=0.03
29	LM004	參數設定：λ=0.04
30	LM005	參數設定：λ=0.05
31	LM006	參數設定：λ=0.06
32	LM007	參數設定：λ=0.07
33	LM008	參數設定：λ=0.08
34	LM009	參數設定：λ=0.09
35	LM010	參數設定：λ=0.10
36	LM020	參數設定：λ=0.20
37	LM090	參數設定：λ=0.90
38	LSI_05	維度：5 維
39	LSI_10	維度：10 維
40	VSM-0	
41	VSM-1	
42	VSM-2	

表 3.6 實驗擷取之特徵總列表

3.5 支援向量機工具及其參數選定與均化步驟

本論文使用 LIBSVM[Chang & Lin 2001]做為實驗工具。LIBSVM 中可以選定參數來達到實驗需求，本論文中設定了表 3.10 所列出之幾種參數進行調整。

參數	說明
-t	調整核心(kernel)，實驗設定核心型態為 RBF(Radial Basis Function)
-w	加重某一類別之權重
-b	產生機率之排序

表 3.7 LIBSVM 參數說明

使用支援向量機必須確保不會有某一特徵的值距和其它的特徵值距差距過大，如此一來容易被此特徵所主導，影響整體的結果。例如，某一特徵t_1的值距介於 1000~2000，另一特徵t_2的值距介於 0.01~0.02 之間，如此一來，就容意被特徵t_1所主導。因此，我們必須對每一特徵做均化的步驟，亦即，對某一特徵中之所有值，使之介於一定範圍之中(如：0~1 之間)。如此一來，特徵與特徵之間的值距就不會差距過大，影響模型的產生。在 LIBSVM 中可以使用 SVM-scale 來達到此種效果，但是，此工具是對每一特徵整體的特徵值進行均化。然而，每一個查詢對應到每一篇文件，皆有一組特徵向量，我們希望此均化的動作，不能跨越查詢，否則會造成實驗數據的錯亂。因此，我們要以每一個查詢所對應到的所有文件之特徵為基準分別進行均化動作，例如，有 16 個查詢時，就必需分別以此 16 個查詢分別對應的文件對進行 16 次的均化動作。在[Liu et al. 2007]中提出一種以查詢為基礎之均化動作，假設對某一查詢i有$N^{(i)}$篇文件，其數學式為式(3.5.1)

$$\frac{x_j^{(i)} - \min\{x_k^{(i)}, k = 1,..., N^{(i)}\}}{\max\{x_k^{(i)}, k = 1,..., N^{(i)}\} - \min\{x_k^{(i)}, k = 1,..., N^{(i)}\}} \qquad (3.5.1)$$

其中x_j^i為一篇文件d_j對應至一個查詢i之其中一個特徵。本論文使用此均化方式對所有特徵進行均化動作。

3.6 支援向量機在資訊檢索之實驗

3.6.1 初步實驗結果

在初步實驗結果中，我們分別以 TDT-2 及 TDT-3 語音正確轉寫文件為檢索目標，使用逐點式訓練中的 SVM 進行訓練，得到的訓練模型作為檢索的模型。而傳統方法中，我們選用 VSM、BM25 及 LM 經由參數設定後，得到的較好檢索結果作為比較之對象。其中，VSM 為特徵名稱 VSM-0 之檢索結果；BM25 為參數設定 $k=0.1$，$b=0.01$ 之檢索結果；LM 為參數設定為 $\lambda=0.1$ 之檢索結果。

■ TDT-2

圖 3.10 為經由 SVM 訓練後，其訓練模型的平均精確率與傳統檢索方法的比較。從圖 3.10 中，我們可以發現，平均精確率僅較 VSM 好，但較 BM25 及 LM 為差。圖 3.11 中呈現的，則是均化遞減累積獲益的結果，經由 SVM 訓練的訓練模型，表現並不理想，在位置 1、位置 3 及位置 5 的檢索效能都最低。因此，我們發現，經過訓練的模型，在 TDT-2 語料中，並沒有得到很好的檢索效能。

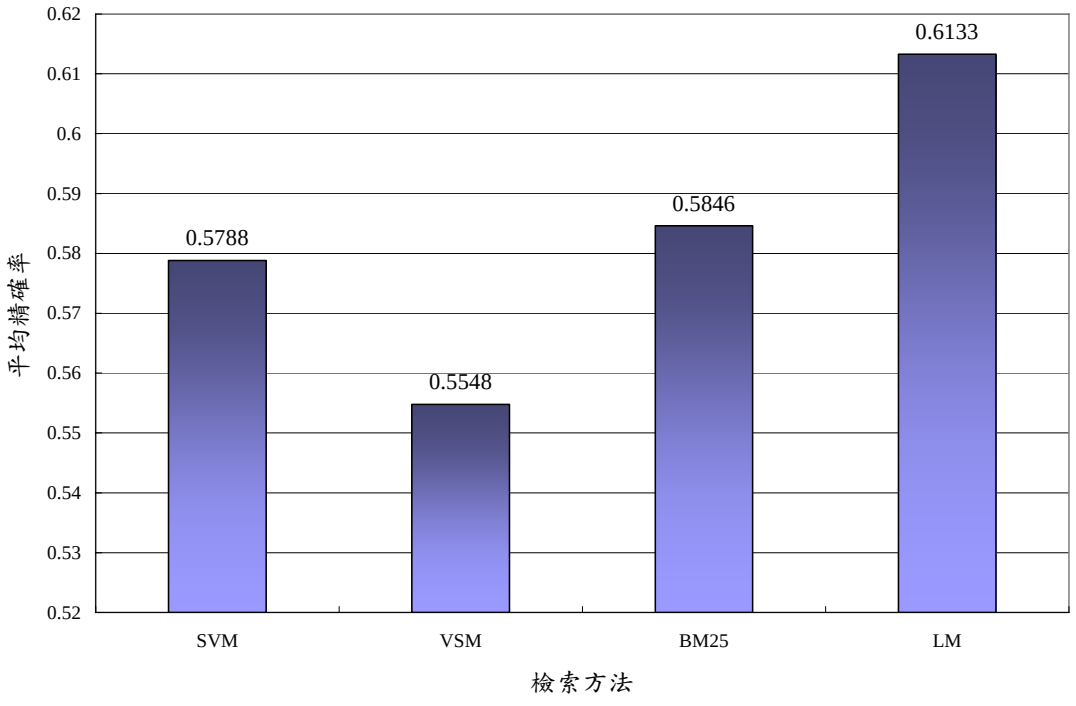

圖 3.10 實驗於 TDT-2 語音正確轉寫文件 SVM 訓練與傳統檢索方法之 MAP

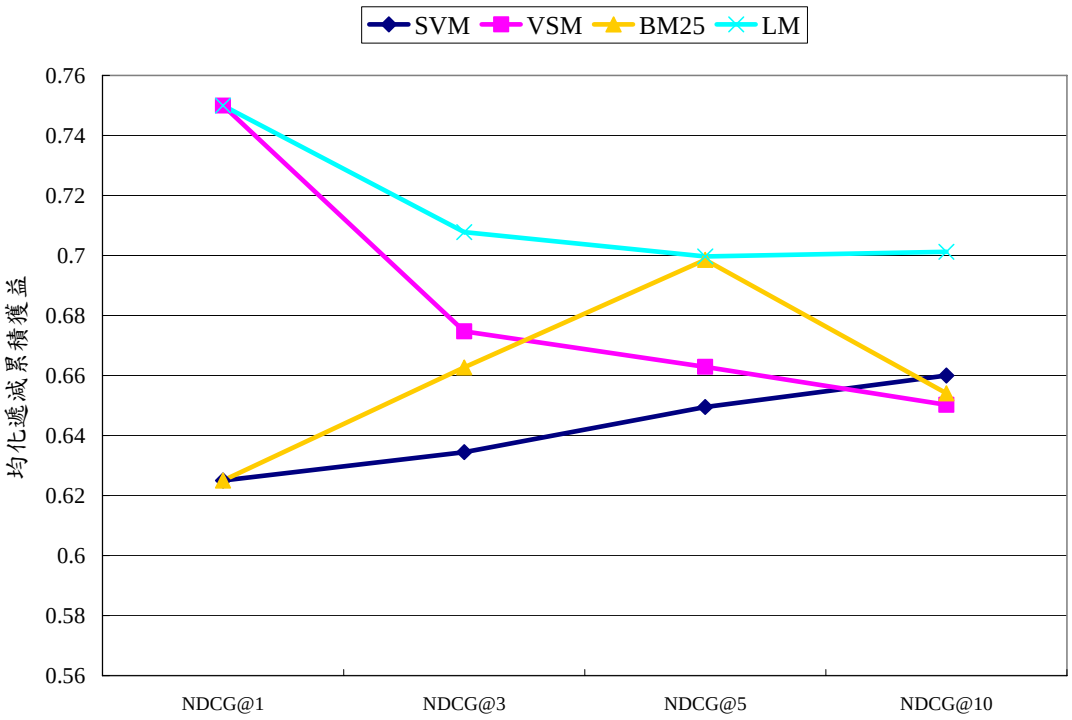

圖 3.11 實驗於 TDT-2 語音正確轉寫文件 SVM 訓練與傳統檢索方法之 NDCG

■ TDT-3

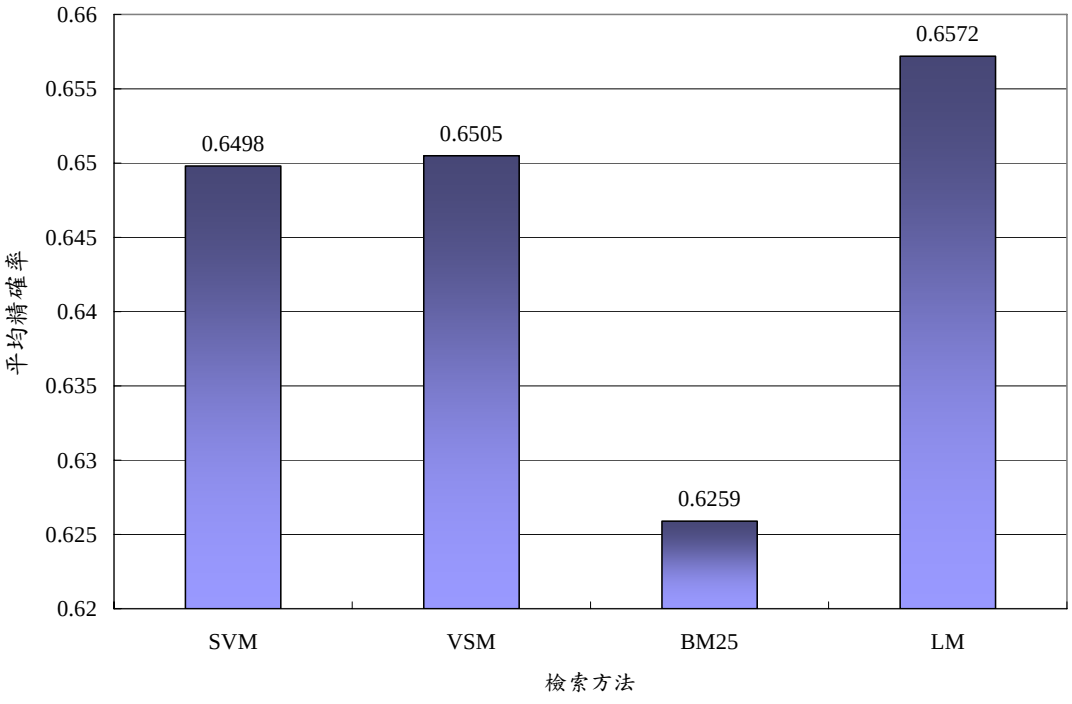

圖 3.12 實驗於 TDT-3 語音正確轉寫文件 SVM 訓練與傳統檢索方法之 MAP

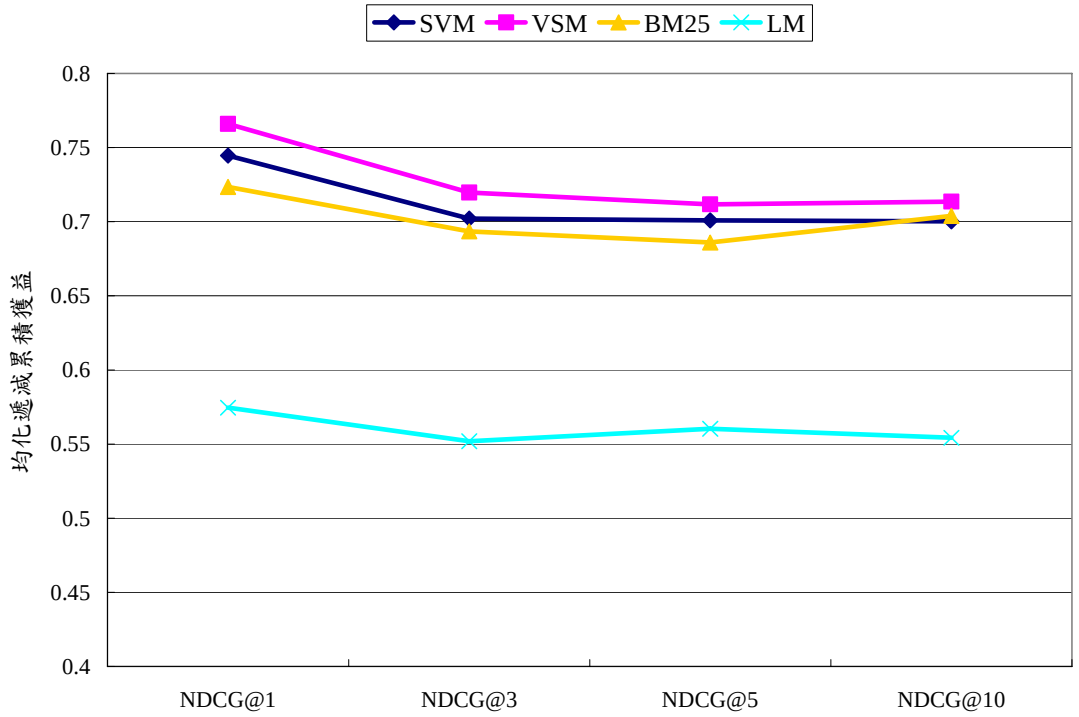

圖 3.13 實驗於 TDT-3 語音正確轉寫文件 SVM 訓練與傳統檢索方法之 NDCG

圖 3.12 為經由 SVM 訓練後，其訓練模型的平均精確率與傳統檢索方法的比較。從圖 3.12 中，我們同樣可以發現，SVM 訓練模型的檢索效能並不理想，平均精確率僅較 BM25 好，但較 VSM 及 LM 為差。圖 3.13 中呈現的，則是均化遞減累積獲益的結果，經由 SVM 訓練的訓練模型，表現並不理想，在各個位置的檢索效能都無法比 VSM 效能為好。因此，我們發現，經過訓練的模型，在 TDT-3 語料中，也沒有得到很好的檢索效能。

3.6.2 問題討論

由初步結果得知，透過支援向量機訓練的訓練模型，相較於傳統檢索方法並沒有得到較好的檢索效能。以下，根據此結果，我們進行研究討論。

由於支援向量機的最佳化評估方式為分類正確率，因此，我們先對平均精確率與支援向量機分類正確率做討論。我們需先觀察在上節中 SVM 之訓練語料分

類正確率與 SVM 之測試語料分類正確率之情形。在 TDT-2 的分類正確情形為表 3.8；在 TDT-3 的分類正確情形為表 3.9，其中精確率(Precision)示意圖如圖 3.14。

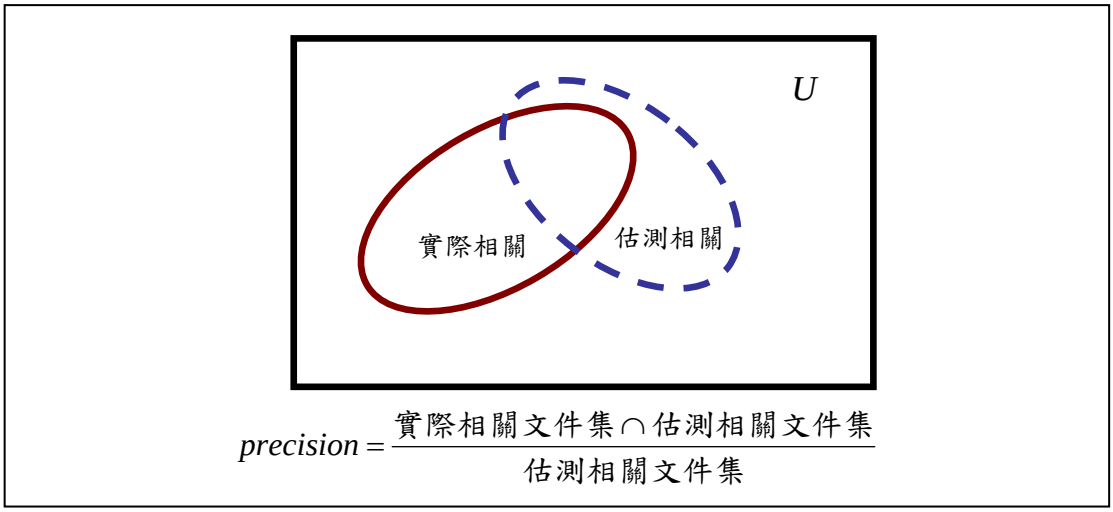

圖 3.13 精確率示意圖

由表 3.8 可知，在 TDT-2 中，雖然支援向量機的測試語料正確率高達 98.8%，但是，其測試語料的精確率卻太低，觀察實際為相關文件的估測錯誤筆數，其錯誤筆數過高。因此，在 TDT-2 中，模型對於實際不相關文件的估測情況良好，但是在實際為相關文件的估測狀況不佳。而同時，我們也發現，不論在訓練語料或是測試語料，實際相關文件的資料數(訓練語料中共有 4931 篇文件，測試語料中共有 468 篇)，遠小於實際為不相關文件的資料數(訓練語料中共有 49310 篇文件，測試語料共有 35872 篇)。因此在訓練時，相關文件的訓練量是比不相關文件的訓練量少很多的。

TDT-2	正確率	Precision	實際為相關文件		實際為不相關文件	
			估測正確	估測錯誤	估測正確	估測錯誤
訓練語料	92.8%	0.4030	1987	2944	48366	944
測試語料	98.8%	0.0833	39	429	35752	120

表 3.8 實驗於 TDT-2 語音正確轉寫文件 SVM 訓練之實驗結果分析

TDT-3	正確率	Precision	實際為相關文件		實際為不相關文件	
			估測正確	估測錯誤	估測正確	估測錯誤
訓練語料	94.5%	0.5927	796	547	13171	259
測試語料	99.5%	0.2500	236	708	157354	139

表 3.9 實驗於 TDT-3 語音正確轉寫文件 SVM 訓練之實驗結果分析

而在表 3.9 中，一樣發現在 TDT-3 語料中，其測試語料正確率高達 99.5%，但是，其精確率依然太低。觀察實際為相關文件的估測錯誤筆數，其錯誤筆數依然過高。因此，在 TDT-2 中，模型對於實際不相關文件的估測情況良好，但是在實際為相關文件的估測狀況不佳。而同時，我們也發現，不論在訓練語料或是測試語料，實際相關文件的資料數(訓練語料中共有 1343 篇文件，測試語料中共有 2287 篇)，遠小於實際為不相關文件的資料數(訓練語料中共有 13430 篇文件，測試語料共有 157493 篇)。因此在 TDT-3 語料中進行訓練時，相關文件的訓練量同樣比不相關文件的訓練量少很多。

總結以上之觀察，我們發現兩種現象：1.當正確率高時，平均精確率仍然不高；2.不論在訓練語料或是測試語料中，皆存在有訓練語料不平衡的問題，亦時，不相關文件的數量遠大於相關文件的數量。以下，我們觀察正確率與平均精確率的關係是否為正相關。倘若正確率與平均精確率為正相關，那麼我們才能確定當正確率變高時，一定能提升平均精確率。然而，在[Yue et al. 2007]中提出了有力的例證中提出了有力的例證，說明了平均精確率與正確率之關係並非正相關。

■ 平均精確率與正確率之比較

平均精確率和正確率並不一定呈現一定的正向的關係，亦即正確率越高不能保證平均精確率也越高。表 3.10(a)為一組 11 則文件的正確相關度解答，經過兩種不同的排序方法，得到了兩組排序結果，如表 3.10(b)，接著，我們可以經由訂定不同的門檻值，得到最佳的正確率。其算法如下：以第一種排序結果而言，

當門檻值定在名次 6 與名次 7 之間時，文件 9、文件 8、文件 7 及文件 6 為實際為相關文件的正確估測，而文件 11 為錯誤估測；文件 5、文件 4、文件 3、文件 2 為實際為不相關文件的正確估測，而文件 1 為錯誤估測，因此，正確估測筆數為 8 筆，可以得到最佳的正確率為 $8/11=0.73$，而第一種排序結果的平均精確率為 $(1/3+2/4+3/5+4/6+5/11)/5=0.51$；第二種排序結果下，當門檻值定在名

文件編號	1	2	3	4	5	6	7	8	9	10	11
正確相關度 相關=1 不相關=0	1	0	0	0	0	1	1	1	1	0	0

(a)

名次	1	2	3	4	5	6	7	8	9	10	11
文件編號 第一種 排序結果	11	10	9	8	7	6	5	4	3	2	1
文件編號 第二種 排序結果	1	2	3	4	5	6	7	8	9	10	11

(b)

	平均精確率	最佳正確率
第一種 排序結果	0.51	0.73
第二種 排序結果	0.56	0.64

(c)

表 3.10 比較平均精確率與正確率之範例

次 9 與名次 10 之間時，可以得到最佳正確率為 $7/11=0.64$，而其平均精確率為 $(1/1+2/6+3/7+4/8+5/9)/5=0.56$。其變化差異可參見表 3.10(c)。由此可見，

當正確率提升時，並不能保證平均精確率也有同樣的提升效果。

對於初步實驗結果不如理想之情況，我們發現 SVM 的訓練模型之訓練依據：正確率，和檢索的評估方法：平均精確率並沒有正相關的關係。此外，我們也發現到訓練語料的相關文件數與不相關文件數比例相差懸殊。但由於 SVM 一直被公認一個效能很好的分類器之一[Manning et al. 2007]。因此，我們並不考量 SVM 本身的模型問題。而造成分類狀況不好，有可能是因為訓練語料相關文件數與不相關文件數比例相差懸殊的特殊語料狀況，亦可能是我們擷取的特徵資訊不足，也最有可能是因為 SVM 並不適合使用於資訊檢索之訓練上。因此，歸納以上所述，可以下三點作為改進目標：

1. 改變訓練模型。
2. 訓練語料資料狀況的改善。
3. 特徵擷取資訊是否足夠。

第四章，我們將對第 1 點及第 2 點進行改進討論。而特徵擷取問題初步不在本論文討論之列。

4. 改進對策

第四章中，我們將針對兩個面向做為改進對策。首先，選擇成對式訓練改變逐點式訓練之訓練語料最佳化評估問題。接著，針對訓練語料中相關文件標籤與不相關文件標籤不平衡問題進行討論，並提出改進對策。

4.1 成對式訓練 - 排序網路(RankNet)

排序網路為一種成對式訓練方法來訓練類神經網路(Neural Network)[Burges et al. 2005]，此方法為每一查詢/文件對組，設計並定義出兩種機率，一是目標機率(Target Probability)，另一種是啟發式機率(Heuristic Probability)。假設我們有兩篇文件d_i與d_j，對某一查詢q而言，目標機率$\overline{P}_{i,j}$定義為

$$\overline{P}_{i,j} = \begin{cases} 1 & ，d_i比d_j相關 \\ 1/2 & ，d_i與d_j同相關 \\ 0 & ，d_j比d_i相關 \end{cases} \tag{4.1.1}$$

啟發式機率則與檢索模型的輸出值有關，若$f(q,d)$是一個以類神經網路為基礎的檢索模型，則啟發式機率$P_{i,j}$定義為

$$P_{i,j} = \frac{e^{f(q,d_i)-f(q,d_j)}}{1+e^{f(q,d_i)-f(q,d_j)}} \tag{4.1.2}$$

排序網路的目標在於期望藉由調整$f(q,d)$內的參數，使得所有$P_{i,j}$能愈靠近$\overline{P}_{i,j}$愈好，為達此目的，排序網路使用了交互熵(Cross Entropy)[Burges et al. 2005]來衡量此二者(目標機率與啟發式機率)距離並做為目標函數(Objective Function)，並使用隨機梯度下降(Stochastic Gradient Descent)來作為最小化的工具。因此，排序網路為每一個文件對組(d_i,d_j)，排序網路定義一個成本函數(Cost Function) $C_{i,j}$:

$$C_{i,j} = -\overline{P}_{i,j}\log P_{i,j} - \overline{P}_{j,i}\log P_{j,i} \tag{4.1.3}$$

經過化簡，可得

$$\begin{aligned} C_{i,j} &\equiv -\overline{P}_{i,j}\log P_{i,j} - \overline{P}_{j,i}\log P_{j,i} \\ &= -\overline{P}_{i,j}\log P_{i,j} - \left(1-\overline{P}_{i,j}\right)\log\left(1-P_{i,j}\right) \\ &= -\overline{P}_{i,j}\log P_{i,j} - \log\left(1-P_{i,j}\right) + \overline{P}_{i,j}\log\left(1-P_{i,j}\right) \\ &= \overline{P}_{i,j}\left[\log\left(1-P_{i,j}\right) - \log P_{i,j}\right] - \log\left(1-P_{i,j}\right) \\ &= \overline{P}_{i,j}\log\left(\frac{1}{P_{i,j}}-1\right) - \log\left(1-P_{i,j}\right) \end{aligned} \tag{4.1.4}$$

上式中的 $P_{i,j}$ 以式(4.1.2)代入式(4.1.4)，可得

$$\begin{aligned} C_{i,j} &= \overline{P}_{i,j}\log\left(\frac{1}{P_{i,j}}-1\right) - \log\left(1-P_{i,j}\right) \\ &= \overline{P}_{i,j}\log\left(\frac{1}{e^{f(q,d_i)-f(q,d_j)}}\right) - \log\left(\frac{1}{1+e^{f(q,d_i)-f(q,d_j)}}\right) \\ &= -\overline{P}_{i,j}\log\left(e^{f(q,d_i)-f(q,d_j)}\right) + \log\left(1+e^{f(q,d_i)-f(q,d_j)}\right) \\ &= -\overline{P}_{i,j}\left(f(q,d_i) - f(q,d_j)\right) + \log\left(1+e^{f(q,d_i)-f(q,d_j)}\right) \end{aligned} \tag{4.1.5}$$

我們分三種情況來討論 $C_{i,j}$：

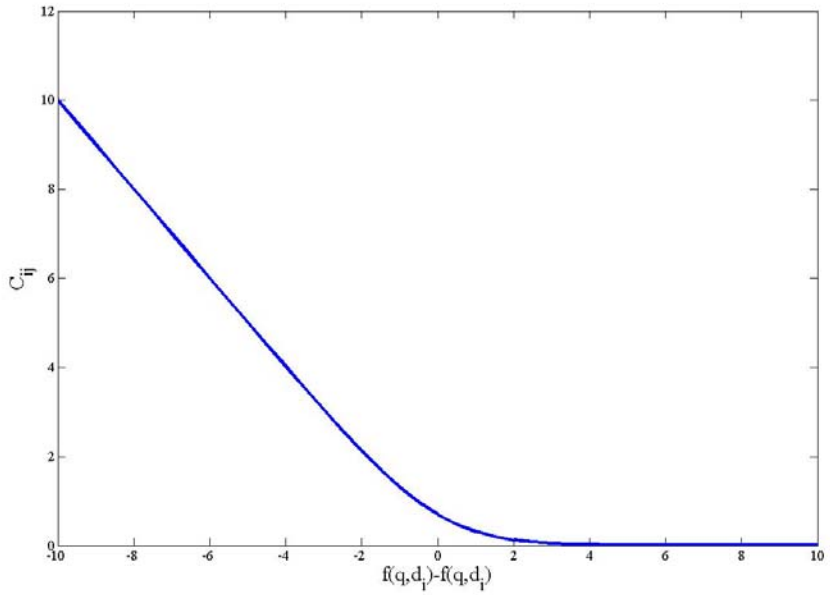

圖 4.1 $\overline{P}_{i,j}=1$ 時， $f(q,d_i)-f(q,d_j)$ 與 $C_{i,j}$ 之關係圖

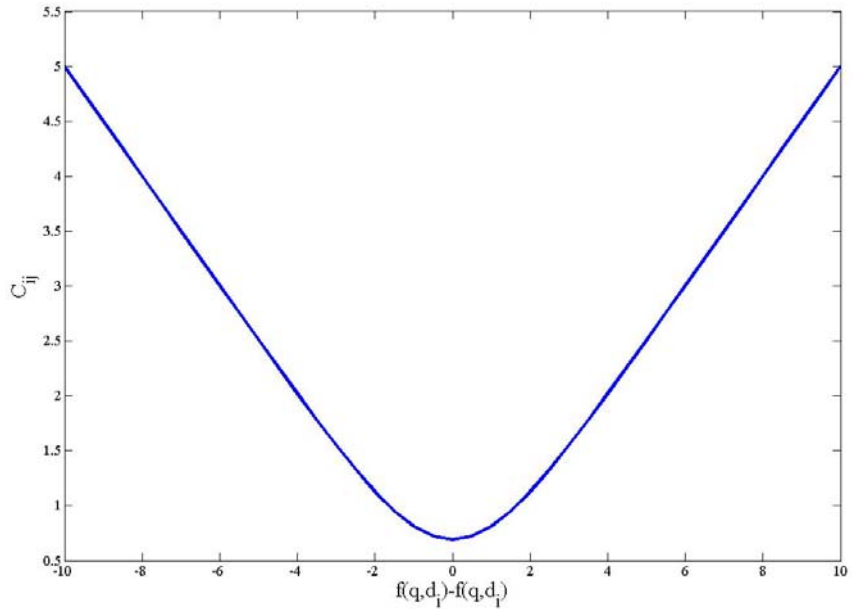

圖 4.2 $\overline{P}_{i,j}=1/2$ 時，$f(q,d_i)-f(q,d_j)$與$C_{i,j}$之關係圖

(1) $\overline{P}_{i,j}=1$，即d_i比d_j更相關，成本函數可化簡成

$$C_{i,j}=-\left(f(q,d_i)-f(q,d_j)\right)+\log\left(1+e^{f(q,d_i)-f(q,d_j)}\right) \qquad (4.1.6)$$

$f(q,d_i)-f(q,d_j)$與$C_{i,j}$的關係如圖 4.1 所示，可知若要$C_{i,j}$愈小，需使$f(q,d_i)-f(q,d_j)$愈大，即拉大$f(q,d_i)$領先$f(q,d_j)$的差距。

(2) $\overline{P}_{i,j}=1/2$，即d_i與d_j同相關，成本函數可化簡成

$$C_{i,j}=-\frac{1}{2}\left(f(q,d_i)-f(q,d_j)\right)+\log\left(1+e^{f(q,d_i)-f(q,d_j)}\right) \qquad (4.1.7)$$

$f(q,d_i)-f(q,d_j)$與$C_{i,j}$的關係如圖 4.2 所示，可知$C_{i,j}$的最小值出現在$f(q,d_i)-f(q,d_j)=0$時，即拉近$f(q,d_i)$與$f(q,d_j)$的差距。

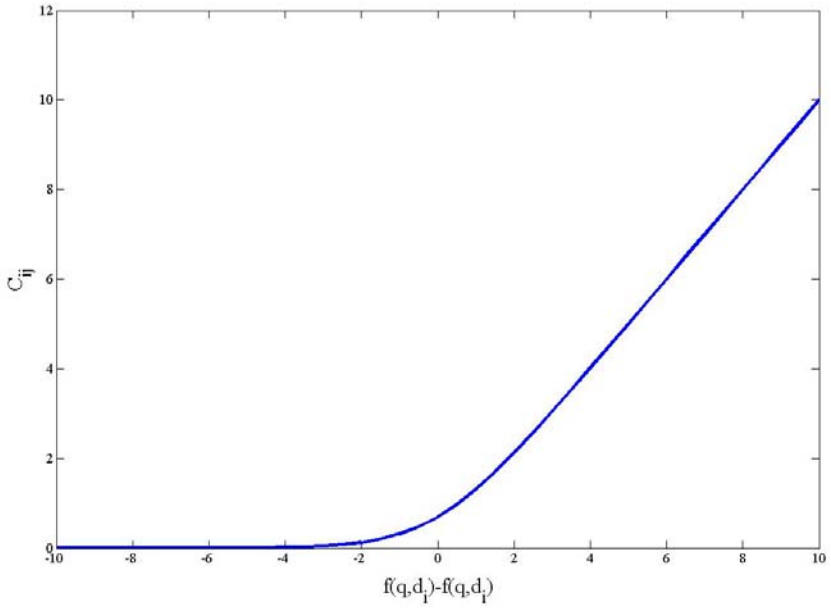

圖 4.3 $\overline{P}_{i,j}=0$ 時， $f(q,d_i)-f(q,d_j)$ 與 $C_{i,j}$ 之關係圖

(3) $\overline{P}_{i,j}=0$，即 d_i 比 d_j 更不相關，成本函數可化簡成

$$C_{i,j}=\log\left(1+e^{f(q,d_i)-f(q,d_j)}\right) \tag{4.1.8}$$

$f(q,d_i)-f(q,d_j)$ 與 $C_{i,j}$ 的關係如圖 4.3 所示，可知若要 $C_{i,j}$ 愈小，需使 $f(q,d_j)-f(q,d_i)$ 愈大，即拉大 $f(q,d_j)$ 領先 $f(q,d_i)$ 的差距。

有許多研究提出不同的成本函數來反應實際上檢索模型的評估效能，都是期望藉由降低定義的成本函數來達到提升評估效能的目的，如真誠度排序演算法(Frank) [Tsai et al. 2007]承襲排序網路但使用真誠度損失(Fidelity Loss)作為成本函數等。為了要最小化成本函數，排序網路承襲類神經網路的隨機梯度降低作為最佳化的方法。

4.2 訓練語料不平衡問題的解決策略

[Nallapati 2004]中提出，資訊檢索是一個具有不平衡訓練集的問題，「不相關文件

對組」亦即「查詢/不相關文件」在整體的資料中佔據著相當大的比重，而「相關文件對組」亦即「查詢/相關文件」則佔據較小的部份。目前對這樣的資料問題的解決方法可分為兩個部份。第一，在一般在支援向量機中解決此種不平衡訓練集的方法，主要是加重佔據比重較少資料的分類正確率結果，可以進行的做法為，將佔據比重較少資料重覆複製為好幾份，拉近與比重較大資料的距離，使之較為平衡。第二，在資訊檢索訓練之中，每一個查詢先對文件集進行排序，排序在前 n 篇者才取之為訓練語料，如圖 4.4。

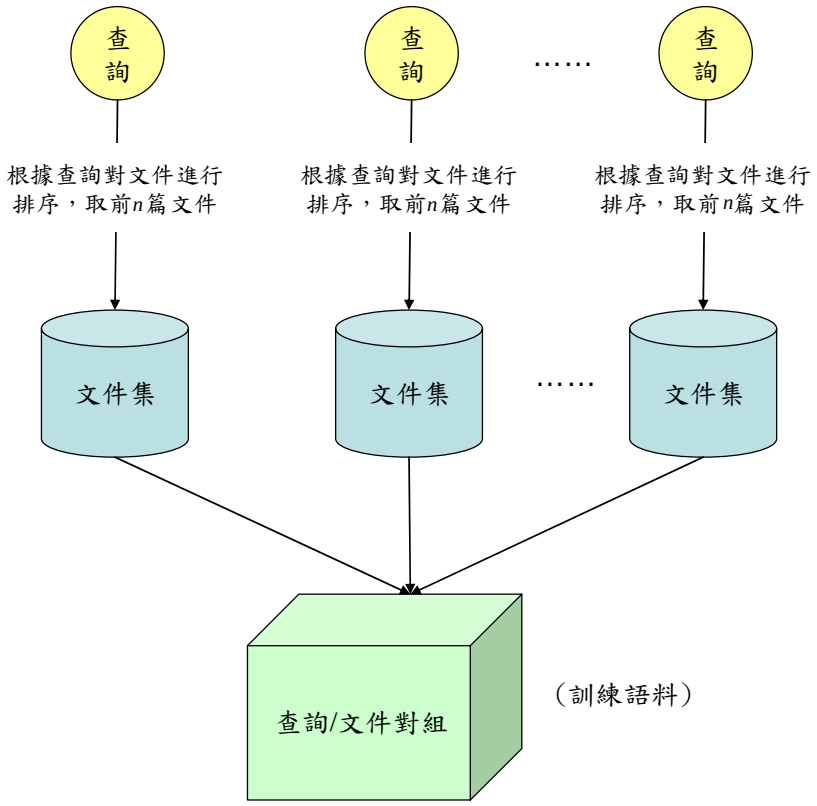

圖 4.4 資訊檢索訓練前處理

然而，上述之兩種方法，正例/反例資料不平衡的問題仍然存在，此問題的主要原因在於查詢/文件對組中，正例的資料數量遠遠低於反例的資料數量，因此，要克服此問題，可從兩個方向出發，增加正例訓練資料的數量與減少反例訓

練資料的數量，我們將分兩小節個別討論：

4.2.1 增加正例訓練資料的數量 (Up-Sampling)

若 $R(\cdot)$ 為一函數，代表查詢與文件之間的相關關係，其關係如式(4.2.1)所示

$$R(q \to d) = \begin{cases} 0, & d\text{為}q\text{的}relevant\ document \\ 1, & Otherwise \end{cases} \tag{4.2.1}$$

我們假設 $R(\cdot)$ 存在下面兩個性質：

性質 1：

對稱性(Symmetric)： $R(q \to d) = 1 \Leftrightarrow R(d \to q) = 1$。

性質 2：

遞移性(Transitive)： $if\ R(q \to d_1) = 1\ and\ R(d_1 \to d_2) = 1 \Rightarrow R(q \to d_2) = 1$ 。

證明：由於 $R(q \to d_1) = 1$，由性質 1 可知 $R(d_1 \to q) = 1$，且 $R(q \to d_2) = 1$，故由性質 2 可推得 $R(d_1 \to d_2) = 1$。同理可證 $R(d_2 \to d_1) = 1$，所以 $R(d_1 \leftrightarrow d_2) = 1$。由此定理可知，若 q 與 d_1 及 d_2 相關，則 d_1 及 d_2 也互為相關「查詢/文件對組」，故我們可以利用此結論來增加正例訓練資料的數量。

■ 有限制的 *k*-means 演算法

k-means[MacQueen 1967]是一個解決分群問題的演算法，其使用於非監督式學習之中，將一個處在 n 維空間中的資料群切分為 k 群。其主要想法為，先定義 k 個質心點(Centroid)，質心點即是分群的起始依據點。質心點的選取必須相當小心，因為不同的質心點，其對應的分群結果也會不相同。接著，將空間中每一個資料點，分別去計量和哪一個質心點最為靠近，將之屬於其質心點所在之類別。當所有的資料點皆分類完畢之後，即結束第一階段的分群。接下來，必須重新選定新的質心點。新的質心點之求法，可以根據第一階段的分群結果，我們擁有 k 個資料群，對每一資料群求資料群之質心(Barycenters)當做新的質心點。擁有新的 k 個質心點，又可以對同樣的所有資料點重新分群。以此方式循環計算，直到前一階段的質心點位置與下一階段的質心點位置距離差在一個門檻值之

D:文件集

必須鏈結集合：$MustLink = \{(d_i, d_j) | \exists C_m, s.t.\ d_i \in C_m \text{ and } d_j \in C_m\} \subseteq D \times D$

不可鏈結集合：$CannotLink = \{(d_i, d_j) | \text{if } d_i \in C_n \text{ then } d_j \notin C_n\} \subseteq D \times D$

Con-Kmeans(D , $MustLink$, $CannotLink$)

1. 初步選定 C_1,..., C_k 之 k 個質心點
2. *if* Violate-Constrains(d_j , C_j , $MustLink$, $CannotLink$)= *false*
   將D 中所有點d_j 根據此k 個質心點指定
   至最相近之群 C_j 。
3. 將每一群C_i 中之所有點 d_j 取平均，當做新的質心點。
4. 重複(2)及(3)直至收斂。
5. 回傳最終之{C_1,..., C_k}

Violate-Constrains(d_j , C , $MustLink$, $CannotLink$)

1. for each $(d_j, d_k) \in MustLink$
   if $d_k \notin C$ return true
2. for each $(d_j, d_k) \in CannotLink$
   if $d_k \in C$ return true
3. otherwise return false

圖 4.5 有限制的 *k*-means 演算法

下，即可停止。*k*-means 分群方法是一個經常被使用的自動化切割資料於 k 群的演算法[Wagstaff et al. 2001]。

分類演算法(Clustering Algorithms)通常如 *k*-means 一般，使用在非監督之情況中[Wagstaff et al. 2001]。亦即，分類演算法通常使用一些特徵來描述資料點，而沒有給定任何其它資訊，例如：標籤(Labels)。然而，在實際的狀況中，部份的資料點有時是會擁有一些背景知識的，例如：部份資料擁有正確之分群類別。在大多數的分類演算法中，即使知道這樣的資料，也無法使用[Wagstaff et al. 2001]。因此，有限制的 *k*-means 方法即是架構在此問題之上所發展出來之概念。

首先，有限制的 *k*-means 在兩兩資料點之間，先定義了兩種限制：必須鏈結(Must-link)以及不可鏈結(Cannot-link)。必須鏈結限制此兩個資料點必須位於同一群；不可鏈結限制此兩個資料點必須不能位於同一群。而必須鏈結具有遞移封閉(Transitive Closure)性質。圖 4.4 中為有限制的 k-means 之演算法。先選定$C_1,...,C_k$之k個質心點，接著，對D中之每一個d_j，決定d_j最接近哪一個質心點，在指定於最接近的C_j之前，必須先判斷：

(1)與d_j有必須鏈結限制且已經經過指定群之所有點d_k是否不屬於C_j，

(2)與d_j有不可鏈結限制且已經經過指定群之所有點d_k是否屬於C_j。

若有以上兩項判斷之任一項符合，則d_j不能被指定至C_j群，必須接著判斷與d_j次接近之C_k群。當D中所有點皆已被指定群之後，分別對$C_1,...,C_k$中之所有點計算新的質心點。依以上之步驟反覆至前一階段之與下一階段之質心點的改變在門檻值之下，即可停止，最後完成之$C_1,...,C_k$即是所須之分群結果。在此演算法中，需留意的是，必須鏈結的所有鏈結須先定義完全，例如：當 *A MustLink B*，*C MustLink B*，則必須將 *A MustLink C* 之條件亦納入必須鏈結集合之中，如此才不會發生分群順序之問題(*A* 先進行分群之結果與 *C* 先進行分群之結果不同)。

4.2.2 減少反例訓練資料的數量 (Down-Sampling)

要減少反例訓練資料的數量，可藉由控制訓練資料中不相關文件的數量來達成，然而要選出具代表的文件群卻是不容易。Brendan J. Frey 教授等人在 2007 年 315 期的「科學」(Science)期刊上提出了一個方法 [Frey & Dueck 2007]，名為關係傳遞(Affinity Propagation)演算法。

- Affinity Propagation

藉由資料點間的互相推舉，可產生具代表性的點(Examplars)，可用來解決此問

題。假設$s(i,k)$代表d_i與d_k之間的相似度，更精確地說，為d_k適合代表d_i的程度，在我們的應用中給定為d_i與d_k的餘弦評估分數；$r(i,k)$為附屬力(Responsibilities)，用來衡量d_i是否認為d_k足以代表d_i；$a(i,k)$為支配力(Availability)，用來衡量d_k對d_i的支配度。$r(i,k)$與$a(i,k)$的更新過程可透過式(4.2.1)及式(4.2.2)之間的交互影響，一開始時，所有的支配力$a(i,k)$均為0，並經由數次的迭代，達到最後的穩定狀態。

$$r(i,k) \leftarrow s(i,k) - \max_{k' \, s.t. \, k' \neq k} \{a(i,k') + s(i,k')\} \tag{4.2.1}$$

$$\begin{cases} a(i,k) \leftarrow \min\left\{0, r(k,k) + \sum_{i' \, s.t. \, i' \notin \{i,k\}} \max\{0, r(i',k)\}\right\} & , i \neq k \\ a(k,k) \leftarrow \sum_{i' \, s.t. \, i' \notin \{i,k\}} \max\{0, r(i',k)\} & i = k \end{cases} \tag{4.2.2}$$

以下我們將以「戰國爭霸」為劇本，分別討論$s(i,k)$、$r(i,k)$及$a(i,k)$，及說明上述兩式的概念：

(1) $s(i,k)$為d_k適合代表d_i的程度：

由於$s(i,k)$自一開始即可自行給定，不會被更新，可視為天生d_i認同d_k當中原霸主的認同度。$s(i,i)$則可解釋為天生的國(武)力強弱，也是一開始就需給定的值，一般來說，對所有d_i，$s(i,i)$可設相同，代表每個國家問鼎中原的起跑線都相同，若不相同，較大$s(i,i)$代表天生有較大的機率當霸主，故也可解釋為(上天)偏好的程度(Preference)。

(2) $r(i,k)$為d_i對d_k的歸屬力、援助能力，由式(4.2.1)更新：

由於戰國時期，國家之間常有衝突，$r(i,k)$可解釋為當d_k要擴張版圖時，d_i可以提供的援助。一般而言，d_i對d_k的援助能力$r(i,k)$，與d_i對d_k的認同度$s(i,k)$成正向關係，由於認同度$s(i,k)$是天生的不會改變，但援助能力卻會因不同的狀況而所改變，如當出現另一個國家$d_{k'}$對d_i有很強的支配力(統制力)時，d_i能給d_k的援助能力$r(i,k)$就會被削弱(因為$d_{k'}$會要求d_i予以援助，故d_i尚需留些餘力來應付$d_{k'}$)，故d_i對d_k的援助能力$r(i,k)$，需再扣除

$\max_{k' s.t. k' \neq k} \{a(i,k') + s(i,k')\}$。而 $r(k,k) \leftarrow s(k,k) - \max_{k' s.t. k' \neq k} \{a(k,k') + s(k,k')\}$，即為 d_k 天生的國(武)力 $s(k,k)$扣除援助他人的能力，最後可解釋為 d_k 的剩餘能力。

(3) $a(i,k)$為 d_k 對 d_i 的支配度，由式(4.2.2)更新：

$a(i,k)$亦可解釋成當 d_k 要對 d_i 開戰時的武力，若我們先忽略 min 及 max 運算，$a(i,k) \leftarrow r(k,k) + \sum_{i' s.t. i' \notin \{i,k\}} r(i',k)$，即 $a(i,k)$等於 d_k 的剩餘能力 $r(k,k)$加上所有 $d_{i'}$ 能給 d_k 的援助 $r(i',k)$。但若 $r(i',k)$為負值怎麼辦？$r(i',k)$為負值代表 $d_{i'}$ 能給 d_k 的援助為負的，表示 $d_{i'}$ 幫倒忙(愈幫愈忙)，不僅不幫忙，反而需要 d_k 提供援助，進而削弱了別國提供的援助，d_k 開戰在即，當然也無瑕兼顧 $d_{i'}$，所以只接受實質的援助(援助為正)，故將各國的援助與 0 取最大值後再相加，即 $\sum_{i' s.t. i' \notin \{i,k\}} \max\{0, r(i',k)\}$。而 d_k 自己對自己的支配力 $a(k,k)$，可解釋為別國援助能力的總和，即 $\sum_{i' s.t. i' \notin \{i,k\}} \max\{0, r(i',k)\}$。

上面說明了 $a(i,k)$可解釋成當 d_k 要對 d_i 開戰時的武力，然而，原作者 Frey 教授等人卻不希望這些國家真的打了起來，若能「和平開戰、禮讓優先」當然最好，為了要推廣這項禮貌運動，原作者設計了一個門檻值來限制武力（支配力），此項限制為：若武力(支配力)比 0 大，則武力(支配力)就歸零。導致支配力 $a(i,k)$大都小於 0，即 d_k 對 d_i 不僅無支配力，反而還是負的，最大也僅僅是零而已。因此大家都變成有禮貌的國家，在選舉的制度中，投票選出中原盟主。

最後，每輪迭代結束，每個國家 d_i 會找出要臣服的國家 $d_k = \arg\max_{k'} \{a(i,k') + r(i,k')\}$，若 $d_k = d_i$，表示 d_i 都認為自己才是霸主，則 d_i 就成了霸主。

【討論】

1. 式(4.2.1)及式(4.2.2)即使做到收斂，仍有可能留下相當多的代表點，代表點的數目即分群的數目需要間接由偏好值(Preference) $s(i,i)$來決定，

$s(i,i)$的大小會決定留下來的代表點的數目，換句話說，偏好值$s(i,i)$與相似度$s(i,k)$的比例會決定分群的數目。

2. 相似度度函數可以是非對稱，即$s(i,k)$與$s(k,i)$可以不同。且可以不遵守三角不等式，故此方法可應用的範圍相當廣泛。

4.2.3 更新方法流程

為了在訓練資料上，有效利用上述增加正例、減少反例的方法，我們設計了一個新的檢索流程如下：

1. 將訓練查詢及文件群一起進行分群，並希望分群過程保持相關的關係，意即：

$$Suppose \quad d_i \in C_x, d_j \in C_y$$
$$if \ R(d_i \leftrightarrow d_j) = 1 \ then \ C_x = C_y$$

2. 為了要在分群時加入限制，我們使用有限制的 *k*-means[Wagstaff et al. 2001] 演算法來分群。若C_i有 *Ns* 篇文件群，則對C_i而言我們可產生$N(N-1)$個正例訓練資料(Positive Training Sample)。
3. 對每一群C_i，利用關係傳遞演算法對C_i取出N_i篇具代表性的文章，故對C_i而言，我們可產生$\sum_{j \neq i} N_i N_j$個反例訓練資料。
4. 對每一群C_i，利用上述方法取得的正反例資料。將所有群的訓練資料集合起來，訓練出一個分類器M。
5. 在測試階段，查詢/文件對組則透過此分類器M來排序。

5. 語音文件檢索

5.1 Dragon 大詞彙語音辨識器

Dragon 大詞彙連續語音辨識器[Zhan et al. 1999]對 TDT 語料中的新聞語音語料提供中文轉寫。其語音辨識率如表 5.1。

正確率(%)		CH	WD
TDT-2	Dragon	81.05	63.33
TDT-3	Dragon	76.95	59.68

表 5.1 Dragon 大詞彙連續語音辨識器之正確率

5.2 臺師大大陸口音中文大詞彙連續語音辨識系統

臺師大大陸口音中文大詞彙連續語音辨識系統，主要區分為前端處理(Front-end Processing)、聲學模型(Acoustic Model)、詞典建立(Lexicon Construction)、語言模型(Language Model)以及詞彙樹複製搜尋(Tree-copy Search)等，章節 5.2.1 至 5.2.4 將分別對這五個部份做簡介。臺師大大陸口音中文大詞彙連續語音辨識系統之正確率如表 5.2。

正確率(%)		CH	WD
TDT-2	TreeCopySearch (2-gram)	61.38	41.71
	WordGraphRescoring (3-gram)	62.29	41.82
TDT-3	TreeCopySearch (2-gram)	57.40	38.61
	WordGraphRescoring (3-gram)	58.48	38.82

表 5.2 臺師大大陸口音中文大詞彙連續語音辨識器之正確率

5.2.1 前端處理(Front-end Processing)

在本論文中使用梅爾倒頻譜係數特徵作為語音訊號的特徵參數。在求取梅爾倒頻

譜係數特徵時，將語音資料切割成一連串部分重疊的音框，每一個音框由 13 維的梅爾倒頻譜係數特徵加上其一階與二階的時間軸導數(Time Derivatives)所形成的 39 維聲學特徵向量所組成。其中 13 維的梅爾倒頻譜係數特徵是由 18 個梅爾頻譜上濾波器組(Filter Banks)的輸出經餘弦轉換求得。同時，為了降低通道效應對語音辨識的影響，在此使用倒頻譜平均消去法(Ceptral Mean Subtraction, CMS)。

5.2.2 聲學模型(Acoustic Model)

本論文使用由左至右連續密度隱藏式馬可夫模型(left-to-right CDHMM)，其中有 112 個右相關(Right-context-dependent, RCD)聲母模型(INITIAL)、41 個前後音不相關(Context-independent, CI)韻母模型(FINAL)及 1 個靜音模型(silence)，共 154 聲韻母個(INITIAL-FINAL)。而每個模型的狀態數分別為 3 至 6 個不等，每一個狀態皆為獨立的高斯混合分布模型(Gaussian Mixture Model, GMM)，其中每個高斯混合分布的個數分別為 1 至 128 個不等。由這 151 個聲韻母模型共可構成 408 個不考慮聲調的基本音節(Toneless Base-syllable)。

5.2.3 詞典建立(Lexicon construction)

由於 Dragon 的詞典取得不易，本論文以 2 萬 4 千詞的 LDC 中文詞典為基礎，再加入由 Dragon 語音辨識器所辨識的文件中抽取出的新詞彙，構成包含了 5 萬 1 千多詞的詞典。

5.2.4 詞彙樹複製搜尋(Tree-copy Search)

本系統的大詞彙連續語音辨識方法是採用由左至右(Left-to-right)、音框同步(Frame-synchronous)的詞彙樹複製搜尋方式 [Aubert 2002]。在詞彙樹中每個分枝(Arc)代表一個 INITIAL 或 FINAL 的隱藏式馬可夫模型，由樹根(Root)到任一個樹梢(Leaf)的路徑代表一個詞或一些發音相同的詞，路徑上的分枝就是代表這個

詞或這些詞會使用到的隱藏式馬可夫模型。具體來說，所採用的詞彙樹複製搜尋演算法，搜尋時每個音框會同時存在數棵詞彙樹複製(Tree Copies)，每個詞彙樹代表不同的語言模型歷史或限制(Language Model History or Constraint)。實際上，搜尋時產生的不完全路徑(Partial Paths)如果擁有相同的語言模型歷史會被歸類在同一棵詞彙樹複製裡，進行隱藏式馬可夫模型狀態層次(State-level)維特比動態規劃搜尋。在每個音框中，若有不完全路徑已抵達樹梢時，代表一個完整詞已可被產生；同時，不同棵詞彙樹複製間已抵達樹梢的不完全路徑，若具有相同的語言模型歷史，則會進行再結合(Recombination)，保留最大分數者，並以它們的語言模型歷史為標註，產生新的一棵詞彙樹複製，或加入到一棵已存在且具有相同語言模型歷史的詞彙數複製中。

值得注意的是，在實作時並不需要真的建立如此多的詞彙樹複製，僅需建立一棵詞彙樹作為搜尋時路徑展開參考之用即可，並分別記錄搜尋時存活下來的隱藏式馬可夫模型狀態節點(也就是不完全路徑目前拜訪到的節點)的相關資訊。另一方面，由於存活的隱藏式馬可夫模型狀態節點可能會隨音框數呈指數倍增加，因此必須以光束剪裁(Beam Pruning) 技術適當地剪裁分數較低的狀態節點或不完全路徑。在執行剪裁動作時會同時考量每一個詞彙樹複製內部狀態節點(Internal Node)下涵蓋的可能拜訪樹梢節點代表之所有詞對應的語言模型機率，並以其中最大者當做每一個詞彙樹複製內部狀態節點的語言模型前看分數(Language Model Look-ahead Score) [Aubert 2002]，再加上內部狀態節點本身搜尋時所累積的解碼分數(Decoding Score)及聲學前看分數 [Chen *et al.* 2004, 2005]當成剪裁比較的依據。

在本系統採用詞單連語言模型前看(Word Unigram Language Model Look-ahead)技術，對每一個詞彙樹複製內部狀態節點，會以其所在分枝(或隱藏式馬可夫模型)之可能拜訪樹梢節點中具最大詞單連語言模型機率，做為該內部狀態節點的語言模型前看分數。此外，在每個音框，會記錄存活的詞彙樹複製樹梢節點中分數較高者的相關資訊(這些樹梢節點本身代表著可能的候選詞)，諸如

它們的語言模型歷史、對應候選詞開始與結束的音框以及搜尋時聲學解碼的分數，然後再依此資訊建立起一個詞圖。並在詞圖上使用更高階的語言模型，如詞三連(Trigram)、詞四連(Fourgram)語言模型等，重新進行一次詞圖動態規劃搜尋(Word Graph Rescoring)，找出最佳的文句。在本論文中，詞彙樹複製搜尋階段是使用詞雙連語言模型，而在詞圖搜尋階段則是使用詞三連語言模型。

5.3 語音文件檢索流程

語音文件的整體檢索流程如圖 5.1。離線處理端，需先對語音文件進行處理。語音文件透過特徵抽取之後，再經過聲學模型及詞典進行語音辨識，得到詞圖(或

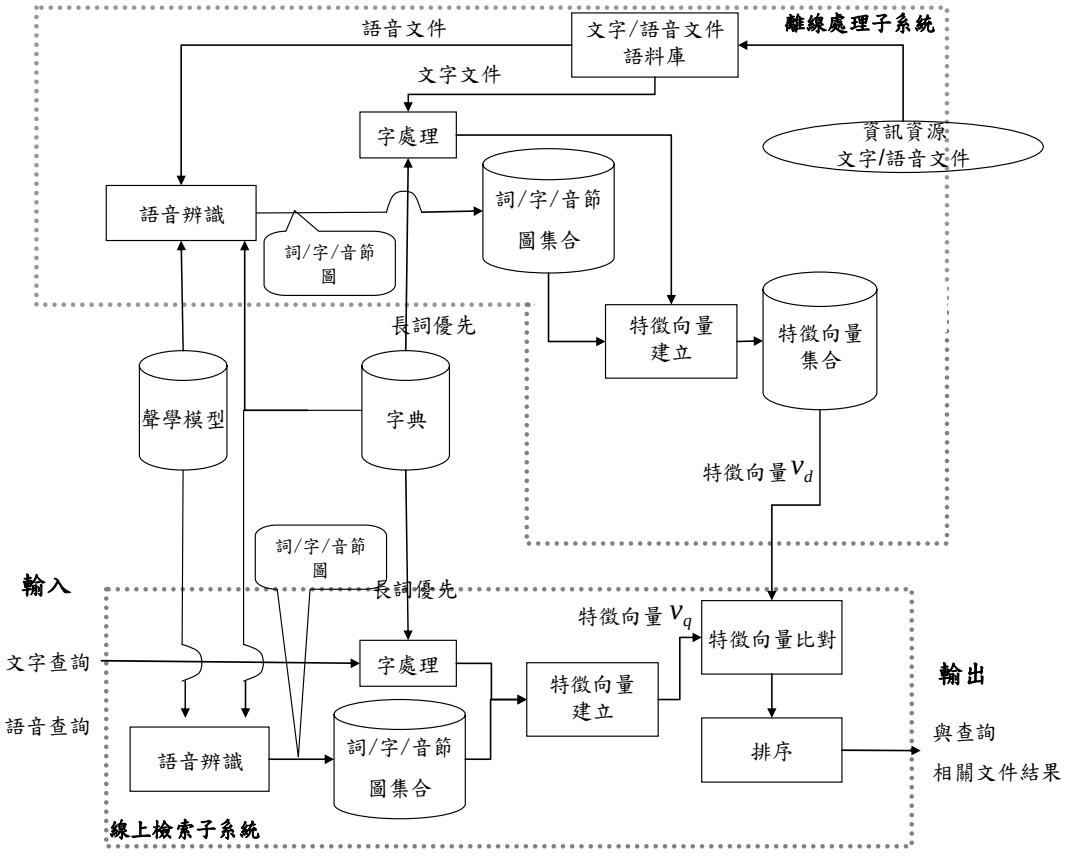

圖 5.1 語音文件的整體檢索流程

是字/音節圖)，此為語音之轉寫文件。對語音轉寫文件建立特徵向量，以供檢索

時比對之用。在線上檢索端，輸入文字或語音查詢，文字查詢經過處理之後，亦能建立查詢之特徵向量；而語音查詢則需經過特徵抽取，將抽取之特徵透過聲學模型及詞典進行辨識，辨識結果為語音查詢之轉寫查詢，即可建立查詢之特徵向量。文件之特徵向量與查詢之特徵向量進行比對及排序，即可得到語音文件之檢索結果。

5.4 個別特徵在語音文件上的檢索效能

對語音自動轉寫文件我們亦抽取與第 3.4 節等同之特徵種類。以下分別對擷取特徵中的相近度模型及機率模型等傳統檢索方法，在 TDT-2 經由 Dragon 辨識器所轉寫之語音文件上，單獨用傳統檢索方法進行檢索之檢索效能。

■ VSM

檢索方法為 VSM 時，其平均精確率結果如圖 5.2。由圖中可發現，，VSM-0 的表現最好，和語音正確轉寫文件上的結果相同。但是，在具有錯誤資訊的語音文件上，不論是 VSM-0、VSM-1 或 VSM-2 都降低了原本的檢索效能。由此可以得

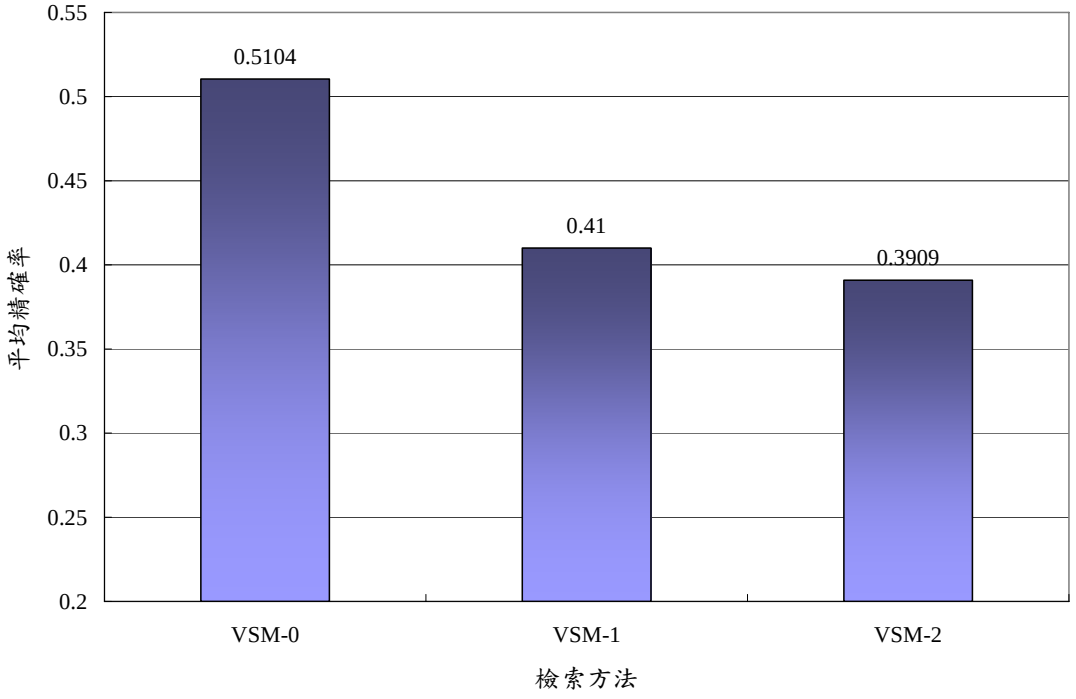

圖 5.2 實驗於 TDT-2 Dragon 辨識器轉寫之語音文件 VSM 之 MAP

知，含有錯誤資訊的語音文件，對檢索效能會有所影響。圖 5.3 為均化遞減累積獲益結果，同樣可以發現，VSM-0 的成效依然最佳。

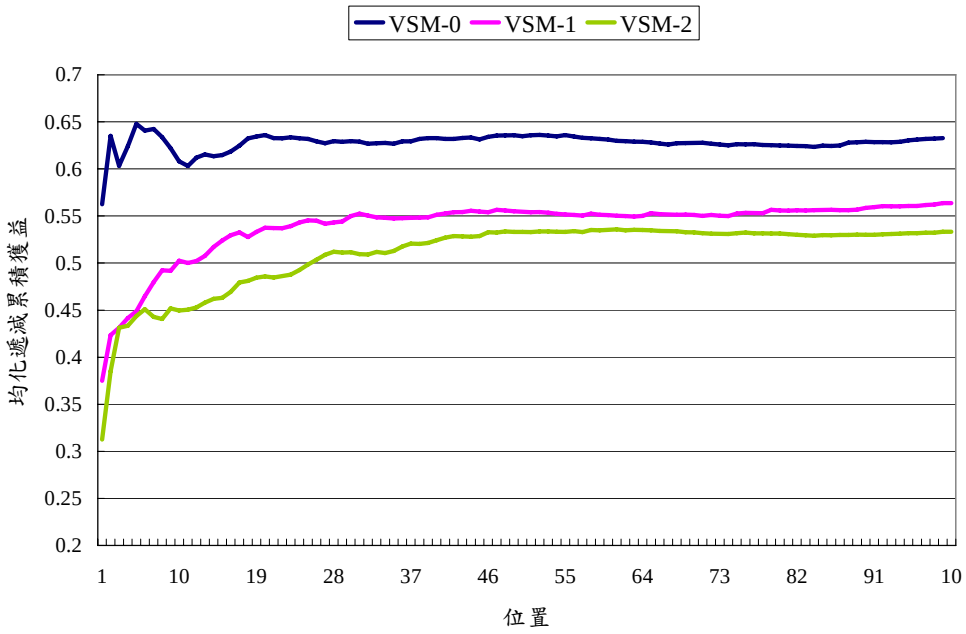

圖 5.3 實驗於 TDT-2 Dragon 辨識器轉寫之語音文件 VSM 之 NDCG

■ BM25

檢索方法為 BM25 時，其各個參數設定下的平均精確率如表 5.3 所示。

由表 5.3 可得知，最佳平均精確率值的參數設定為 k =0.1，b = 0.01。而平均精確率的結果亦較實驗於 TDT-2 語音正確轉寫文件上為低。含有錯誤資訊的語料，同樣影響了 BM25 的檢索成效。圖 5.4 為均化遞減累積獲益之結果，同樣在參數設定為 k =0.1，b = 0.01時成效最好。

模型名稱	*k*	*b*	MAP
BM25_01_001	0.1	0.01	0.5490
BM25_05_001	0.5	0.01	0.4652
BM25_10_001	1.0	0.01	0.4360
BM25_01_050	0.1	0.50	0.5094
BM25_01_005	0.1	0.05	0.5383
BM25_010_010	0.1	0.10	0.5369
BM25_20_075	2.0	0.75	0.3248

表 5.3 實驗於 TDT-2 Dragon 辨識器轉寫之語音文件 BM25 各種參數設定的 MAP

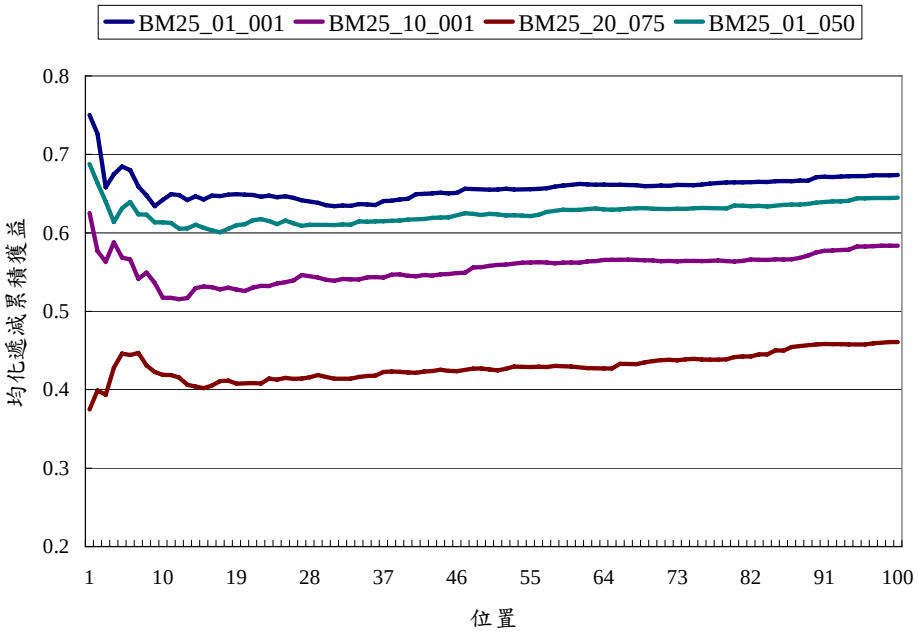

圖 5.4 實驗於 TDT-2 Dragon 辨識器轉寫之語音文件 BM25 各種參數調整下之 NDCG

■ LSI

在 LSI 檢索方法中，圖 5.5 為在不同維度設定下，平均精確率的變化曲線。可以看出，當維度越大時，平均精確率有明顯的下降。維度越大，代表所採用的資訊

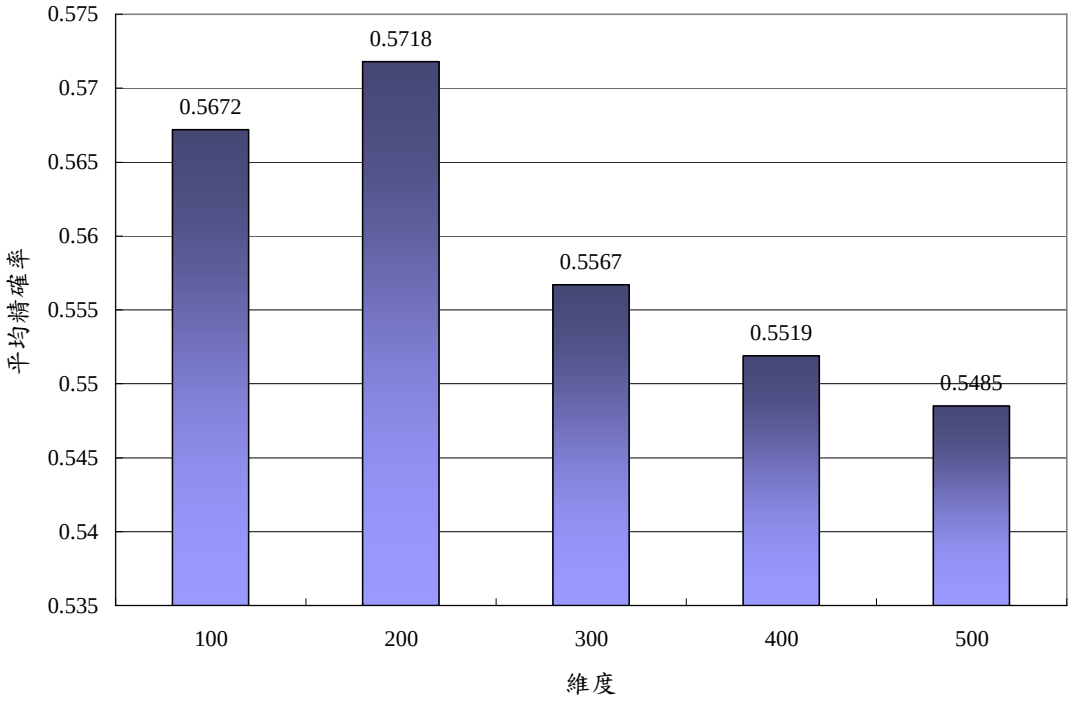

圖 5.5 實驗於 TDT-2 Dragon 辨識器轉寫之語音文件 LSI 各個維度的 MAP

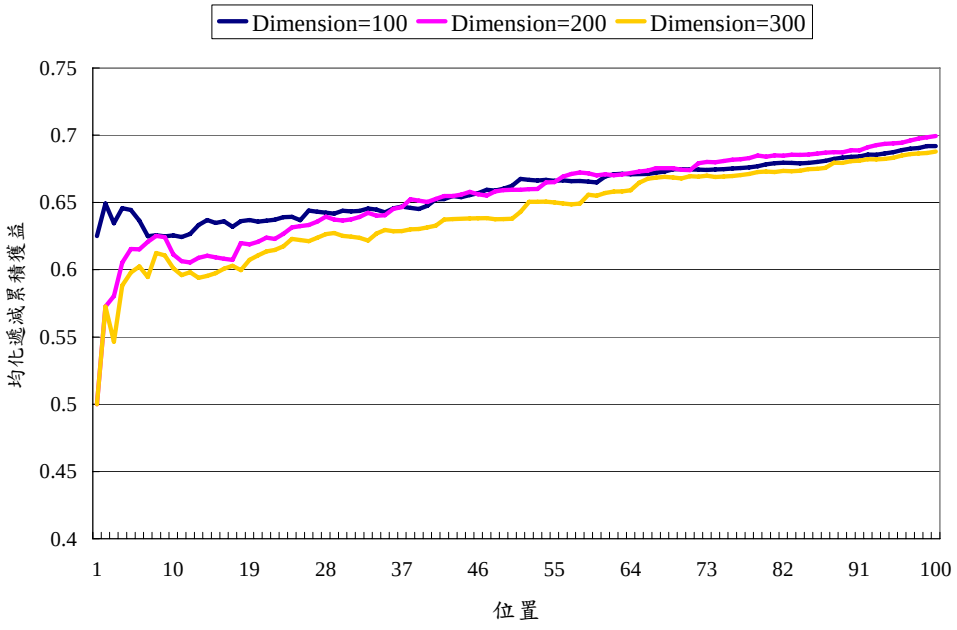

圖 5.6 實驗於 TDT-2 Dragon 辨識器轉寫之語音文件 LSI 各個維度的 NDCG

越多，那麼在含有錯誤資訊的語音轉寫文件上，錯誤量也會提升，因此，平均精確率的效能會下降。而在圖 5.6 中，當維度為 100、200 及 300 時檢索序列中各個位置的均化遞減累積獲益。這和語音正確轉寫文件的結果不同，在語音正確轉寫文件中，維度為 300 時表現的整體均化遞減累積獲益最佳。然而，從圖 5.6 中，我們發現，在維度為 300 時，其表現較維度 100 及維度 200 為差。維度 300 時，錯誤的資訊影響到了資訊檢索的結果，而檢索效能因而下降。

■ LM

表 5.4 為 LM 在參數λ的調整下，不同的平均精確率結果。可以發現，在參數設定λ=0.09 時的平均精確率最高。此平均精確率結果一樣低於在語音正確轉寫文件上的檢索效能。而在圖 5.7 中，均化遞減累積獲益相較於語音正確轉寫文件的結果，依然因文件錯誤率，而降低了檢索效能。

模型名稱	λ	MAP	模型名稱	λ	MAP
LM001	0.01	0.4510	LM007	0.07	0.5354
LM002	0.02	0.5058	LM008	0.08	0.5461
LM003	0.03	0.5180	LM009	0.09	0.5462
LM004	0.04	0.5253	LM010	0.10	0.5456
LM005	0.05	0.5206	LM020	0.20	0.5415
LM006	0.06	0.5306	LM090	0.90	0.4345

表 5.4 實驗於 TDT-2 Dragon 辨識器轉寫之語音文件 LM 各種參數設定下的 MAP

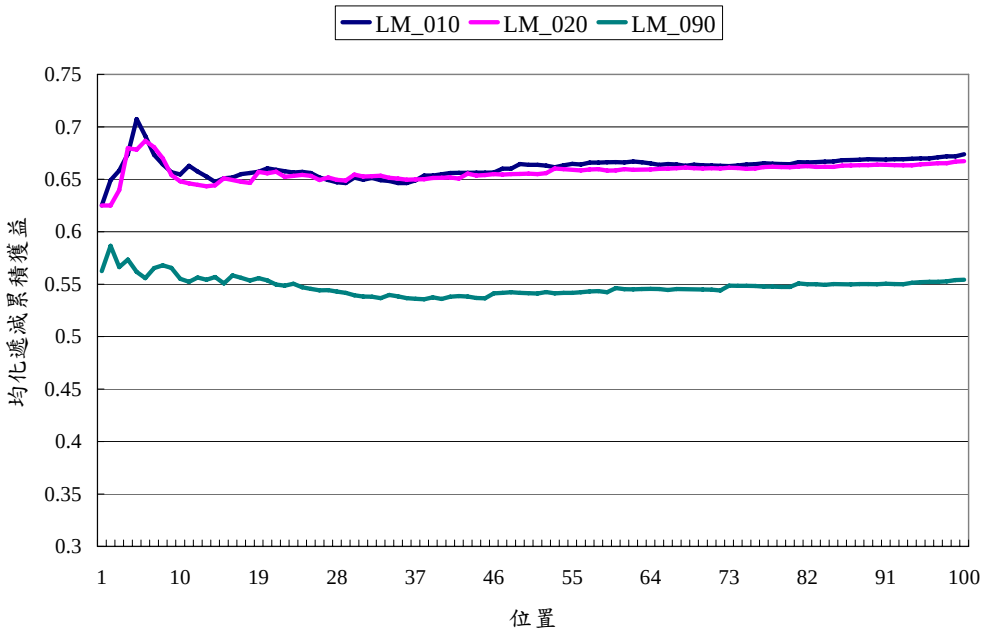

圖 5.7 實驗於 TDT-2 Dragon 辨識器轉寫之語音文件 LM 各種參數設定下的 NDCG

6. 實驗結果與討論

本章實驗包含了四小節，6.1 節探討選用逐點式訓練中的 SVM 進行訓練後，分別在兩種不同辨識率下的語音文件，其各別的檢索效能與傳統資訊檢索方法的檢索效能之比較；6.2 節探討選用成對式訓練中的 RankNet 進行訓練，同樣分別在兩種不同辨識率的語音文件中進行檢索，討論 RankNet 訓練後的檢索效能與傳統資訊檢索方法的效能之比較；6.3 節討論成對式訓練與平均精確率之關係；6.4 節則討論處理不平衡語料問題之結果。在本章中所比較的各項傳統資訊檢索方法，都是經由各種不同的參數嘗試後，取較好的參數設定結果進行比較；BM25 之參數設定：$k=0.1$，$b=0.01$；LM 之參數設定為：$\lambda=0.1$。

6.1 逐點式訓練在語音文件上的檢索

6.1.1 SVM 在 Dragon 語音辨識器轉寫之語音文件的檢索效能

此節中，我們討論使用 Dragon 語音辨識器分別轉寫 TDT-2 及 TDT-3 語音文件，再經由逐點式訓練中的 SVM 進行訓練後之檢索效能。

■ TDT-2

圖 6.1 為實驗於 TDT-2 之平均精確率結果，圖 6.2 為實驗於 TDT-2 之均化遞減累積獲益結果。在 TDT-2 語料中，我們發現平均精確率以 BM25 表現最好，經由 SVM 訓練後的平均精確率僅較 VSM 高，高出 0.0221。在均化遞減累積益中，同樣以 BM25 及 LM 表現較好，而 SVM 訓練卻較 VSM 不理想。這樣的結果顯示，雖然 SVM 訓練後的總體精確率較 VSM 好，但僅看前 10 個排序位置時的相關文件正確率卻較 VSM 差。經過 SVM 訓練之前，我們所擷取的特徵包含了傳統資訊檢索方法的各種參數設定結果，包括參數設定後，檢索效能表現較好的，或者參數設定後，檢索效能不佳者。經由 SVM 訓練之後，我們發現，即使採用了傳統資訊檢索結果的分數作為特徵值之一，訓練後的模型並不能保證其檢索效能一

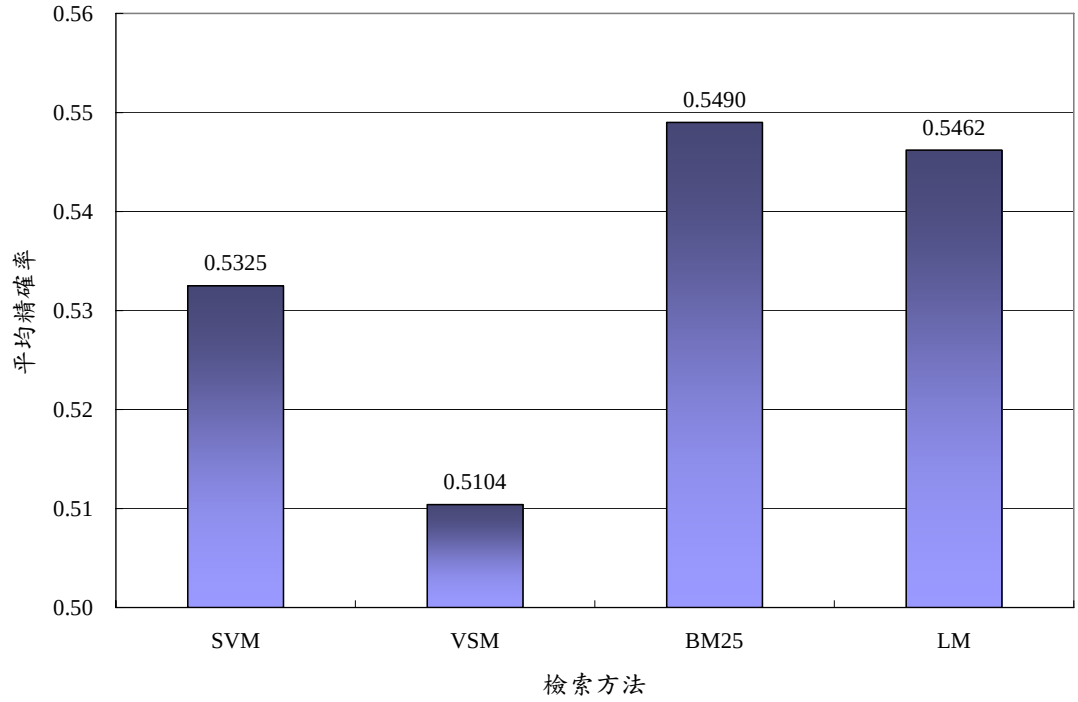

圖 6.1 檢索方法在 TDT-2 使用 Dragon 語音辨識器轉寫之平均精確率

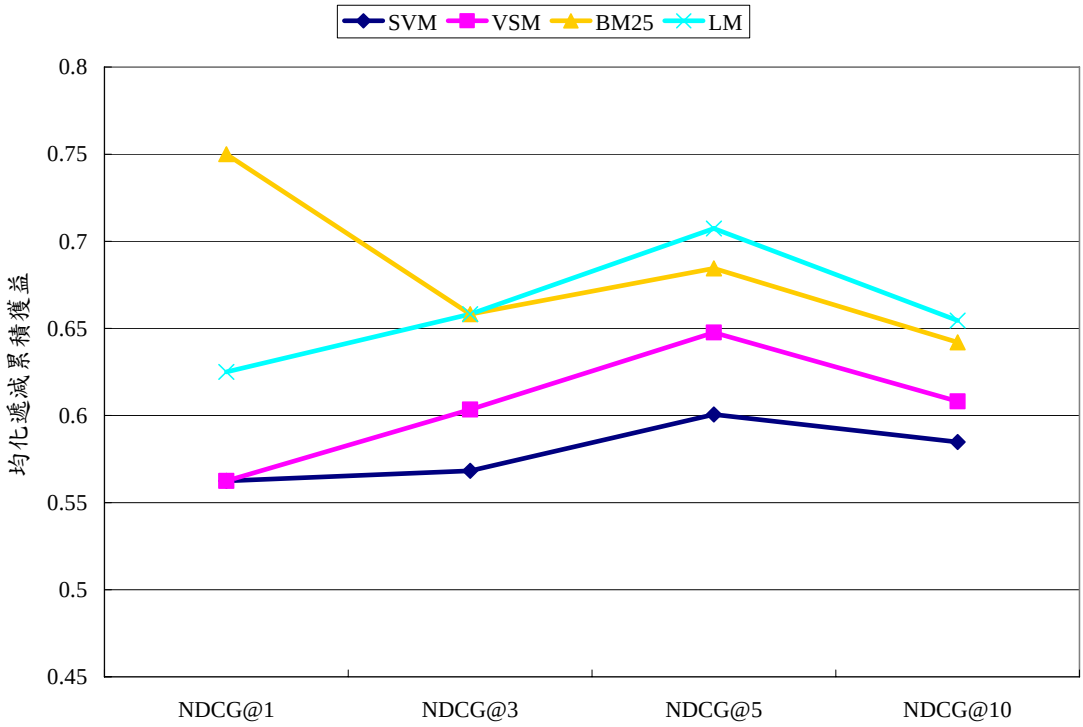

圖 6.2 檢索方法在 TDT-2 使用 Dragon 語音辨識器轉寫之均化遞減累積獲益

定能夠架構在最佳的傳統資訊檢索方法之上。

■ TDT-3

圖 6.3 為實驗於 TDT-3 之平均精確率結果，圖 6.4 為實驗於 TDT-3 之均化遞減累積獲益結果。由圖 6.3 之實驗結果得知，經由 SVM 訓練之檢索結果，其平均精確率效果最佳，為 0.6613。此平均精確率大於傳統資訊檢索方法 VSM：0.0381；大於 BM25：0.0825；大於 LM：0.029。

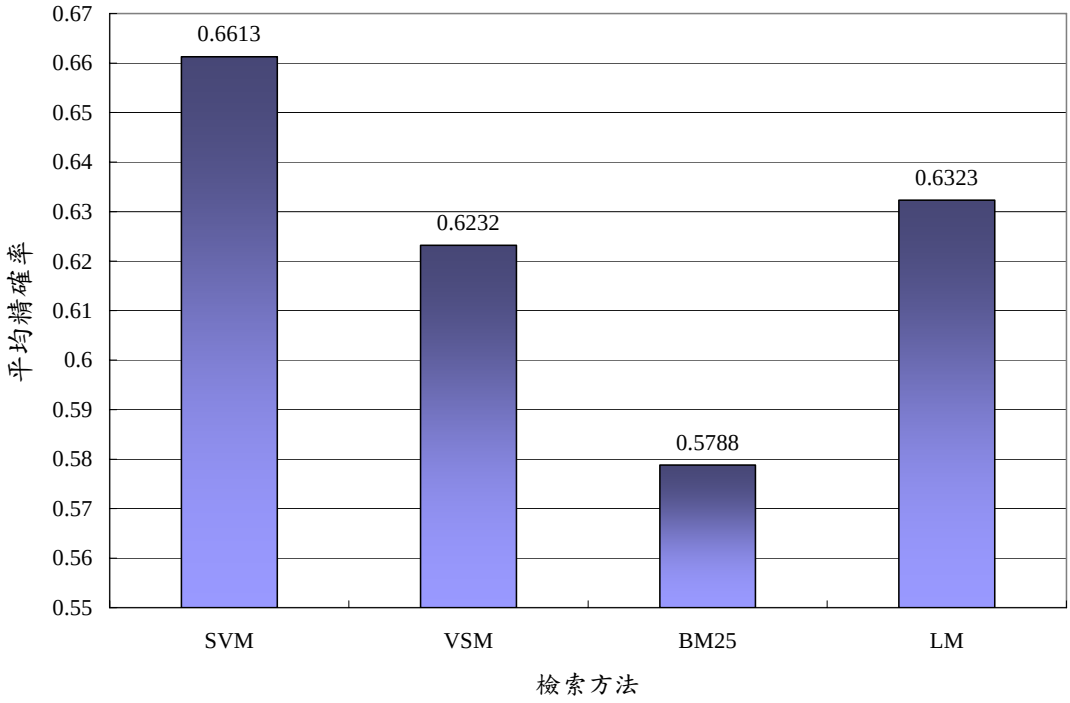

圖 6.3 檢索方法在 TDT-3 使用 Dragon 語音辨識器轉寫之平均精確率

而在圖 6.4 中，經由 SVM 訓練之檢索結果，其均化遞減累積效益也是表現最好。

	NDCG@1	NDCG@3	NDCG@5	NDCG@10
SVM	-	-	-	-
VSM	0.0212	0.0237	0.0150	0.0279
BM25	0.0851	0.0751	0.0429	0.0718
LM	0.0425	0.0289	-0.0014	0.0260

表 6.1 TDT-3 中 SVM 與各項傳統資訊檢索方法在 NDCG 差異狀況

在表 6.1 中可以看出，在各個均化遞減累積效益位置點上，經由 SVM 訓練的效果大都較傳統檢索方法為好。雖然在位置 5 時，LM 略勝於 SVM，但其它位置點，SVM 的效果都較好。

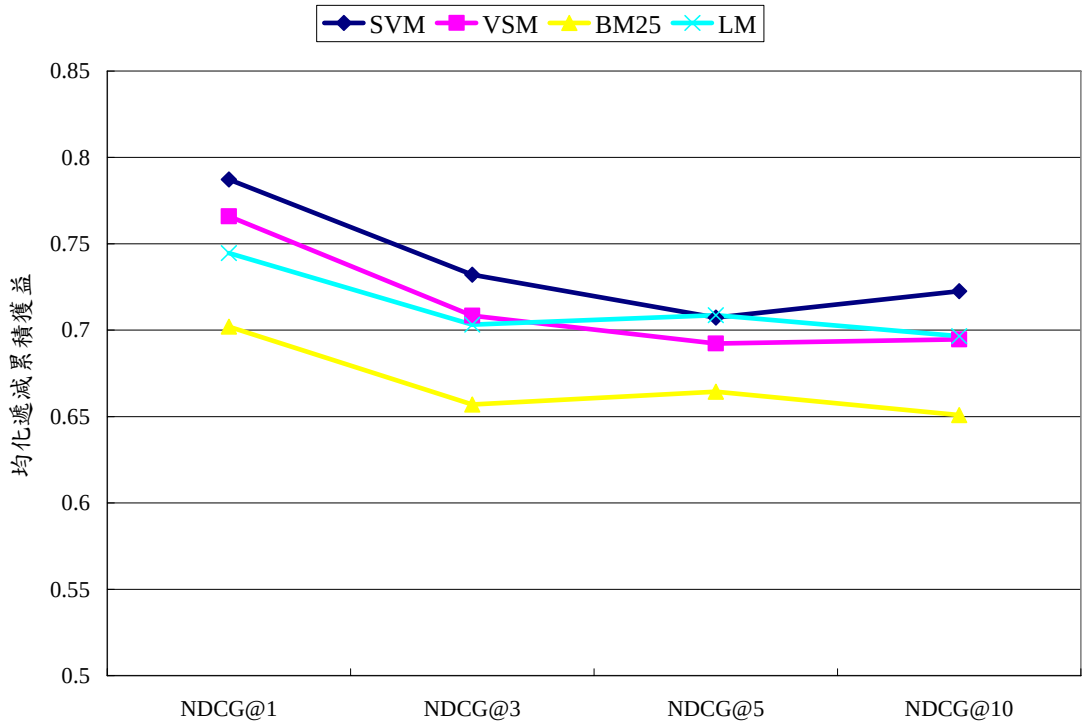

圖 6.4 檢索方法在 TDT-3 使用 Dragon 語音辨識器轉寫之均化遞減累積效益

從以上兩個語料中，我們發現，經由 SVM 訓練後的結果，在 TDT-2 時表現並不理想，但是在 TDT-3 時表現卻最為突出。在這樣的結果之下，我們觀察兩種語料在訓練模型之前，所擷取的各項特徵之差異情形。主要由兩個方向進行觀察：1. 各個特徵的平均精確率；2. 各個特徵彼此之間的排序差異。各別特徵的平均精確率為圖 6.5，此平均精確率是有被選取為訓練資料點才進行計算，並不是整體訓練語料的平均精確率。從圖 6.5 中，我們可以看出，在兩種不同的語料中，所有特徵的平均精確率曲線趨勢類似，在 TDT-3 語料中，各特徵的平均精確率變化幅度較大，而 TDT-2 則較小。接下來，我們觀察特徵彼此之間的排序差異，由於，這是更為精確的比較兩種特徵之間的排序結果，需要細部考量到特徵之間

每一則文件的排序差異。而比較兩種排序結果的方法有許多種，我們選用

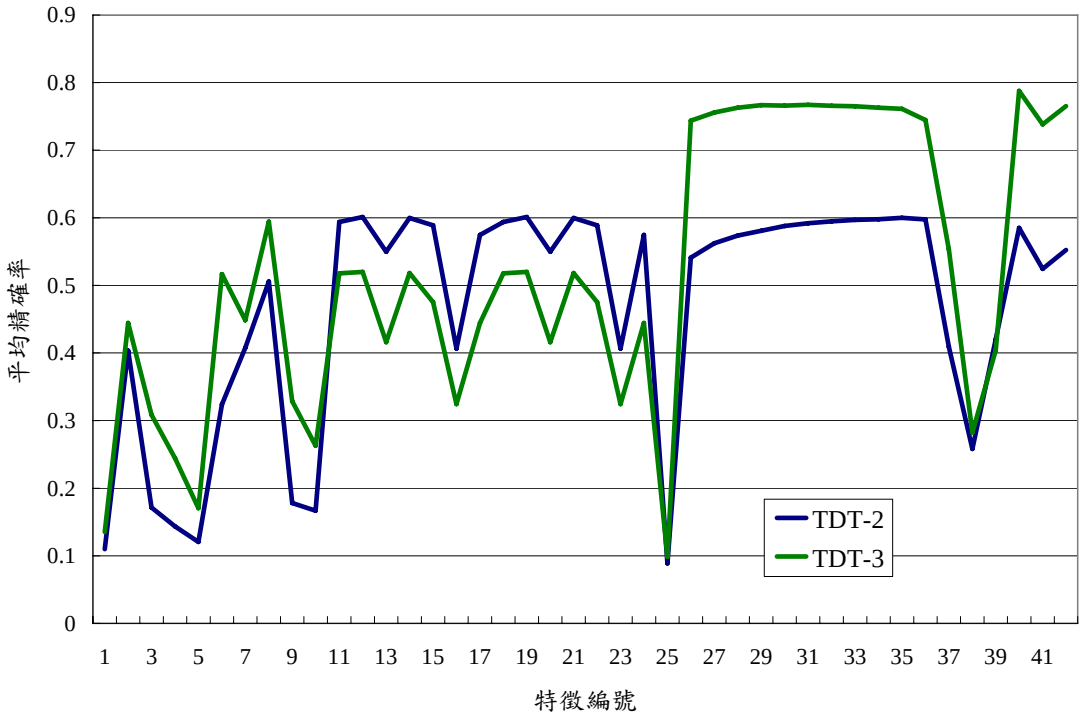

圖 6.5 TDT-2 與 TDT-3 Dragon 辨識器轉寫之訓練語料擷取的各項特徵之 MAP

Spearman's Footrule Distance [Kendall & Gibbons 1990]，來比較每個特徵的排序結果。Spearman's Footrule Distance 如式(6.1.1)所示。

$$d = \sum_{n=1}^{N} |x_n - y_n| \qquad \begin{array}{l} x \in X, y \in Y, \\ X, Y \in \textit{Rank List} \end{array} \tag{6.1.1}$$

Spearman's Footrule Distance 中 X 和 Y 分別代表比較的兩個排序結果。取在 X 序列中的某個文件，其在 X 序列的位置為 x_n，在 Y 序列中排序的位置為 y_n，將此兩個位置資訊相減，並取絕對值。將所有文件位置差異進行加總，就可以得知兩個序列的排序情形。Spearman's Footrule Distance 越小，則代表 X 序列與 Y 序列越相近，Spearman's Footrule Distance 越大，則代表 X 序列與 Y 序列差異較大。圖 6.6 及圖 6.7 分別為 TDT-2 及 TDT-3 的訓練語料經過 Dragon 辨識器轉寫的文件，擷取的所有特徵彼此之間的 Spearman's Footrule Distance。其中，顏色越黑

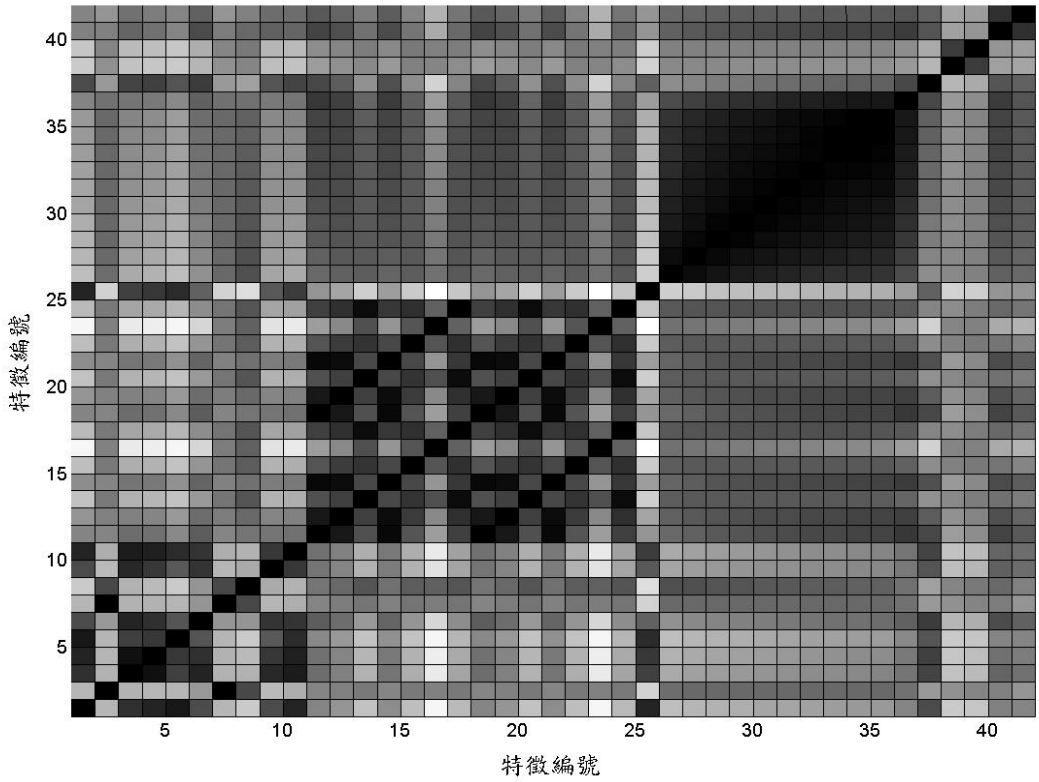

圖 6.6 TDT-2 Dragon 辨識器轉寫之訓練語料特徵之 Spearman’s Footrule Distance

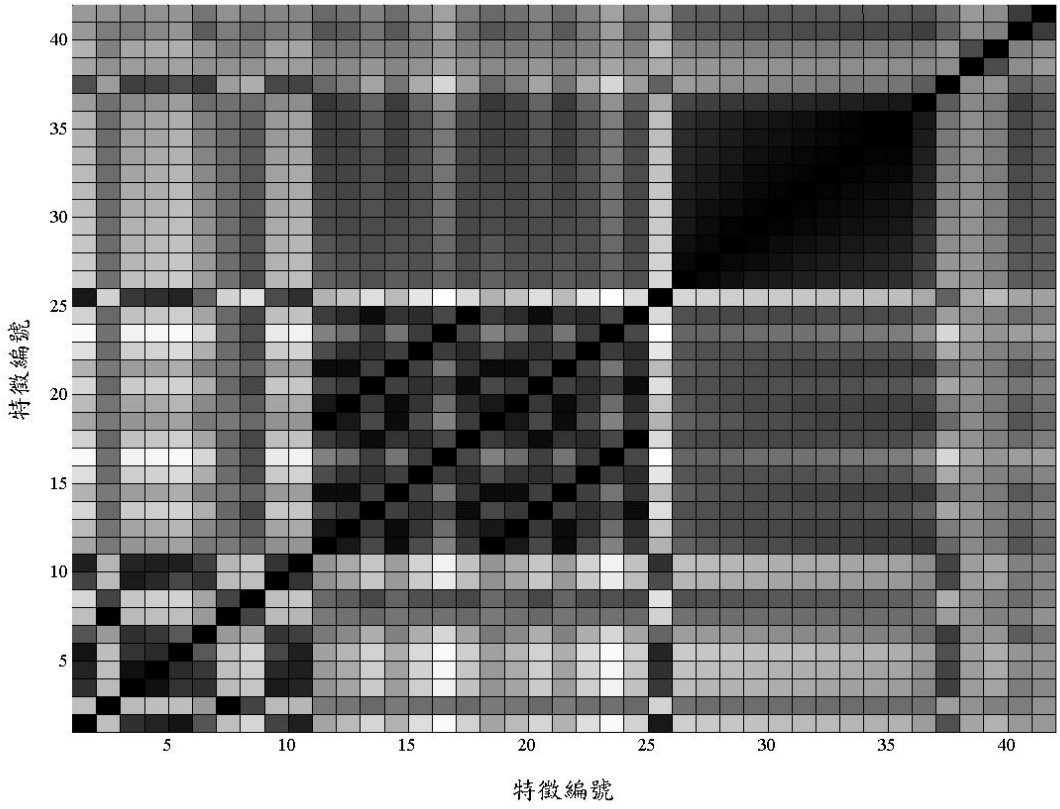

圖 6.7 TDT-3 Dragon 辨識器轉寫之訓練語料特徵之 Spearman’s Footrule Distance

的部份，代表其排序差異大；顏色越淺的部份，代表其排序差異較小。因此，我們可以發現，不論是哪一種語料，低階特徵與其它的特徵排序差異皆較大。TDT-2 與 TDT-3 的差異情形其實很類似，但 TDT-3 的部份差異結果較 TDT-2 顯明。

觀察了訓練時各項特徵的呈現狀況後，我們再來觀察實際的分類情形，為表 6.2 及表 6.3。

TDT-2	正確率(%)	Precision	實際為相關文件		實際為不相關文件	
			估測正確	估測錯誤	估測正確	估測錯誤
訓練語料	93.32%	0.4571	2254	2677	48366	944
測試語料	98.03%	0.4893	229	239	35298	474

表 6.2 TDT-2 Dragon 辨識器轉寫之語音文件 SVM 訓練實驗結果分析

TDT-3	正確率(%)	Precision	實際為相關文件		實際為不相關文件	
			估測正確	估測錯誤	估測正確	估測錯誤
訓練語料	94.65%	0.5920	795	548	13187	243
測試語料	99.44%	0.4566	431	513	157123	370

表 6.3 TDT-3 Dragon 辨識器轉寫之語音文件 SVM 訓練實驗結果分析

從表 6.2 及表 6.3 中，我們可以發現，在訓練語料中，TDT-3 的 SVM 訓練模型的正確率及精確度較 TDT-2 的 SVM 訓練模型為高；在測試語料中，TDT-2 的測試結果精確率其實比 TDT-3 測試結果更好。就這樣的結果中，我們懷疑，其實 TDT-2 的訓練模型的整體分類結果並不差，但在更細度的排序上並沒有很好。因此，我們額外進行了另一個小實驗。首先，我們相信經由 TDT-2 的 SVM 訓練模型後，測試結果大略的排序狀況，依此排序，我們選用 VSM 檢索方法對此排序做細部的調整。也就是說，我們將 SVM 的測試結果之排列序列切分為 n 等份，對此 n 等份的每一等份，內部以 VSM 的序列結果為依據進行細部調整。其示意圖如圖 6.8。接著，我們觀察經過微調之後的序列結果的平均精確率。在圖 6.9 中，比較在 $n=40$ 及 $n=30$ 時與不經過細部處理的平均精確率。我們可以發現，經由細部

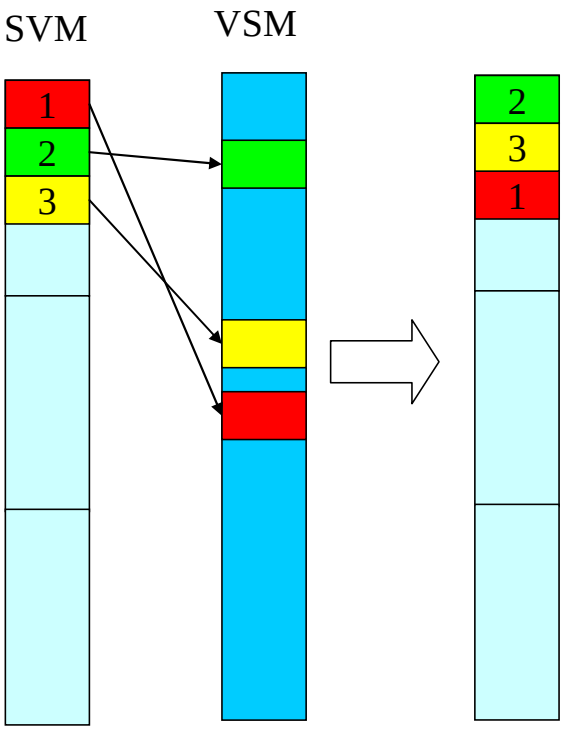

圖 6.8 TDT-2 Dragon 辨識器轉寫之語音文件調整細部排序示意圖

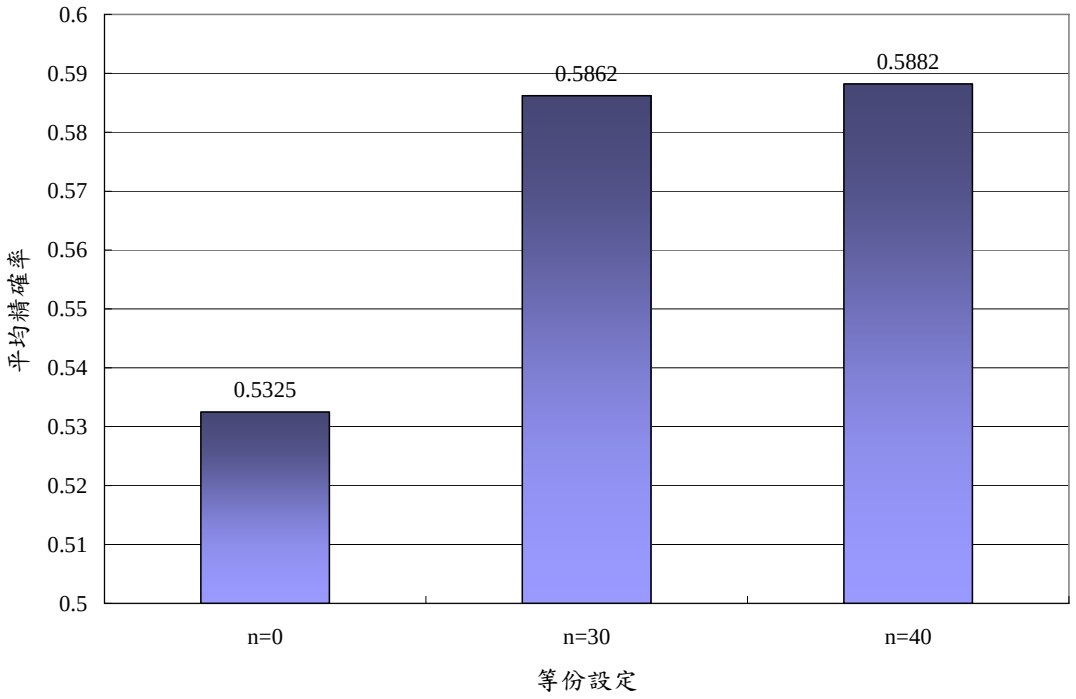

圖 6.9 TDT-2 Dragon 辨識器轉寫之語音文件細部調整排序後之平均精確率

調整之後，平均精確率得到很大的改善。因此，在 TDT-2 中，平均精確率的表現不佳，可能是因為細部的排序並不好，但整體而言，SVM 的分類狀況並不差。

6.1.2 SVM 在臺師大大陸口音中文大詞彙語音辨識器轉寫之語音文件的檢索效能

此節中，我們討論由臺師大大陸口音中文大詞彙語音辨識器所辨識之語音轉寫文件使用逐點式訓練中的 SVM 進行訓練後之模型的檢索效能。

- TDT-2

圖 6.10 及圖 6.11 分別為 TDT-2 經由臺師大大陸口音中文大詞彙語音辨識器轉寫後之各項檢索方法的平均精確率結果及均化遞減累積獲益結果。由圖 6.10，在 TDT-2 時經由 SVM 訓練後，在較低辨識率轉寫文件下的平均精確率，相較於傳統資訊檢索方法，效果最好。然而，由圖 6.11 亦可得知，在前 10 個位置下的均化遞減累積獲益，經由 SVM 訓練之模型，其表現卻不是最佳。因此，就整體的相關正確率而言，經由 SVM 訓練之模型成效最好，但僅就部份前 10 個排序的相關正確率，並不是最理想。

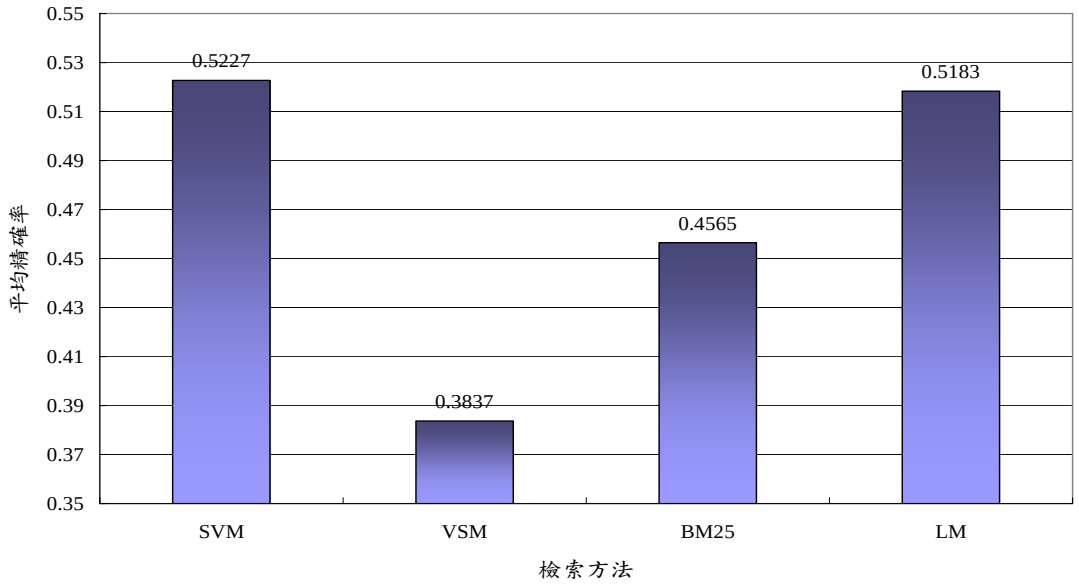

圖 6.10 檢索方法在 TDT-2 臺師大大陸口音中文大詞彙語音辨識器轉寫之 MAP

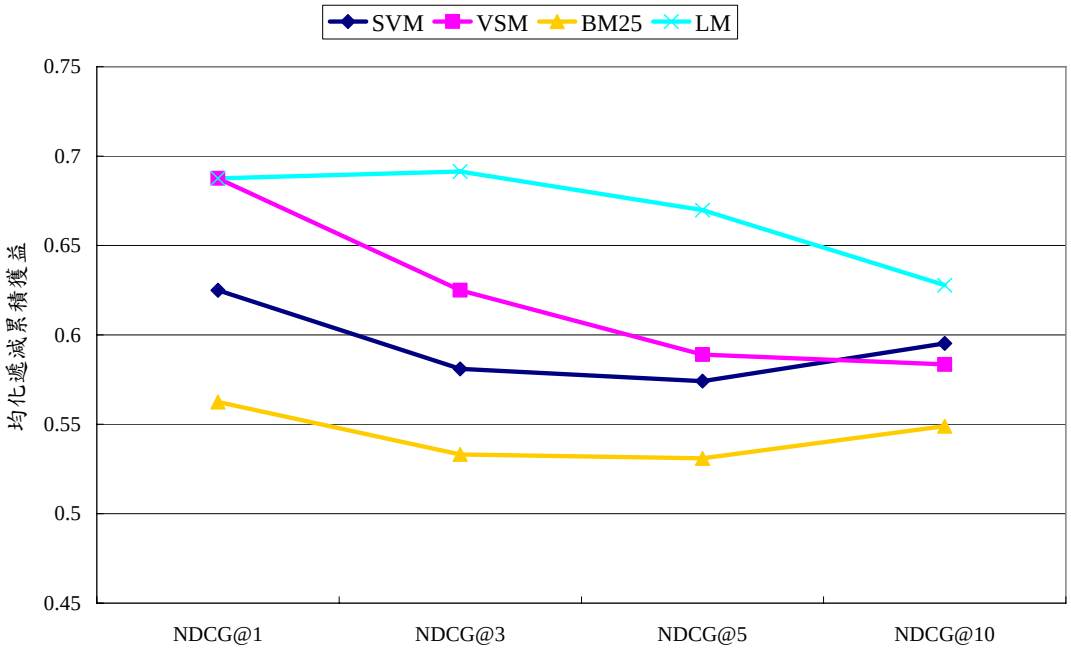

圖 6.11 檢索方法在 TDT-2 使用臺師大大陸口音中文大詞彙語音辨識器轉寫之 NDCG

■ TDT-3

圖 6.12 及圖 6.13 分別為 TDT-3 經由臺師大大詞彙語音辨識器轉寫後，各項檢索

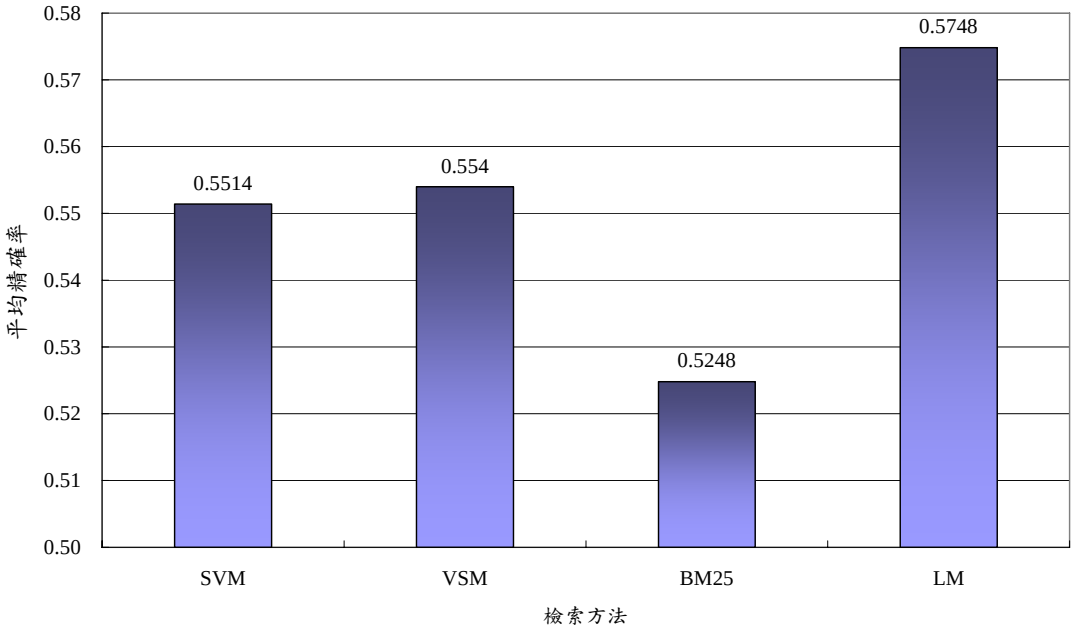

圖 6.12 檢索方法在 TDT-3 使用臺師大大陸中文大詞彙語音辨識器轉寫之 MAP

方法的平均精確率結果及均化遞減累積獲益結果。由圖 6.12，在 TDT-3 語料中，經由 SVM 訓練後，在較低辨識率轉寫文件下的平均精確率，相較於傳統資訊檢索方法，效果並不理想。由圖 6.11 亦可得知，在前 10 個位置下的均化遞減累積獲益，經由 SVM 訓練之模型，儘在位置 5 時表現最好，其它的位置點皆不盡理想。

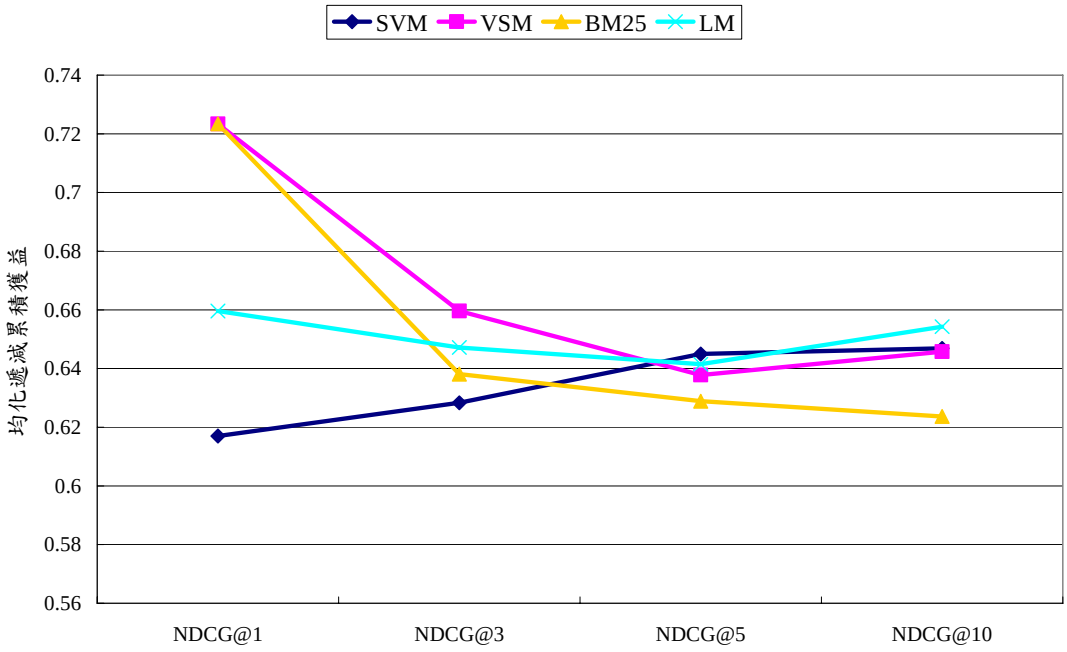

圖 6.13 檢索方法在 TDT-3 使用臺師大大陸口音中大詞彙語音辨識器轉寫之 NDCG

由以上的結果，我們依然可以發現，在使用臺師大大陸口音中文大詞彙語音辨識器轉寫後的文件中，使用逐點式訓練的 SVM 進行訓練後，實驗於 TDT-2 的成效較好，但實驗於 TDT-3 的成效較不理想，因此，我們也觀察兩種語料在訓練模型之前，所擷取的各項特徵之差異情形。同樣由兩個方向進行觀察：1. 各個特徵的平均精確率；2. 各個特徵彼此之間的排序差異。各個特徵的平均精確率為圖 6.14 所示。由圖 6.14 可以發現，TDT-2 和 TDT-3 各特徵的平均精確率曲線趨勢一致，而 TDT-3 的特徵間的平均精確率差異較 TDT-2 為大。圖 6.15 及圖

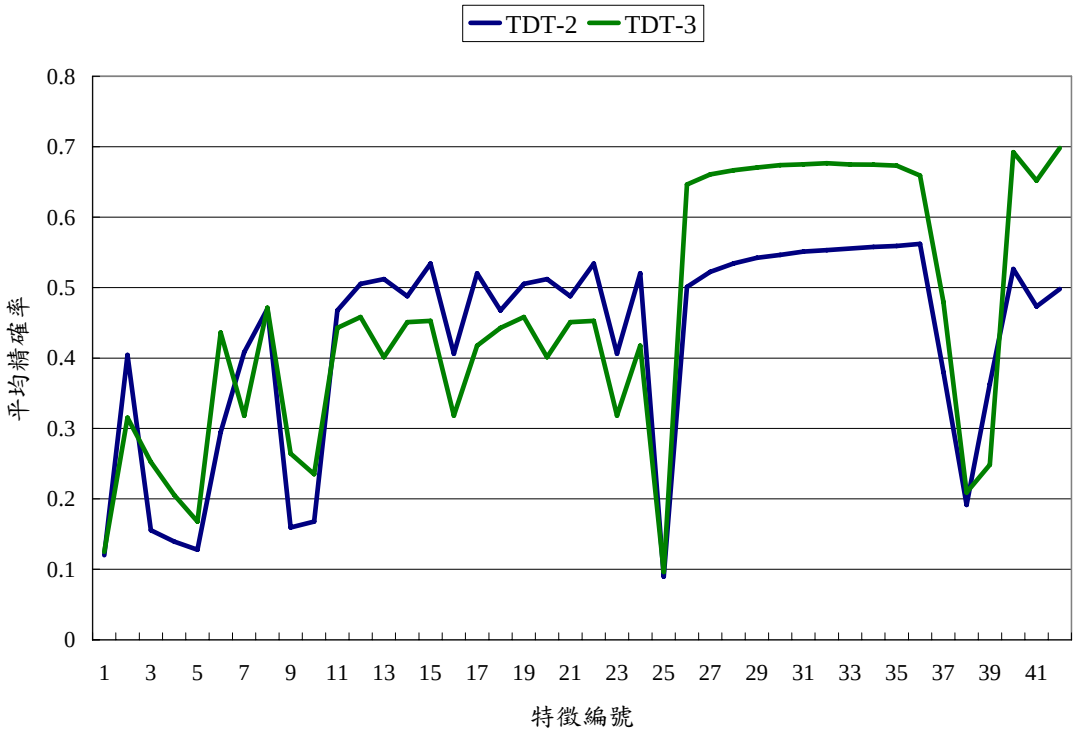

圖 6.14 TDT-2 與 TDT-3 使用臺師大大陸語音中文大詞彙語音辨識器轉寫之訓練語料擷取的各項特徵之 MAP

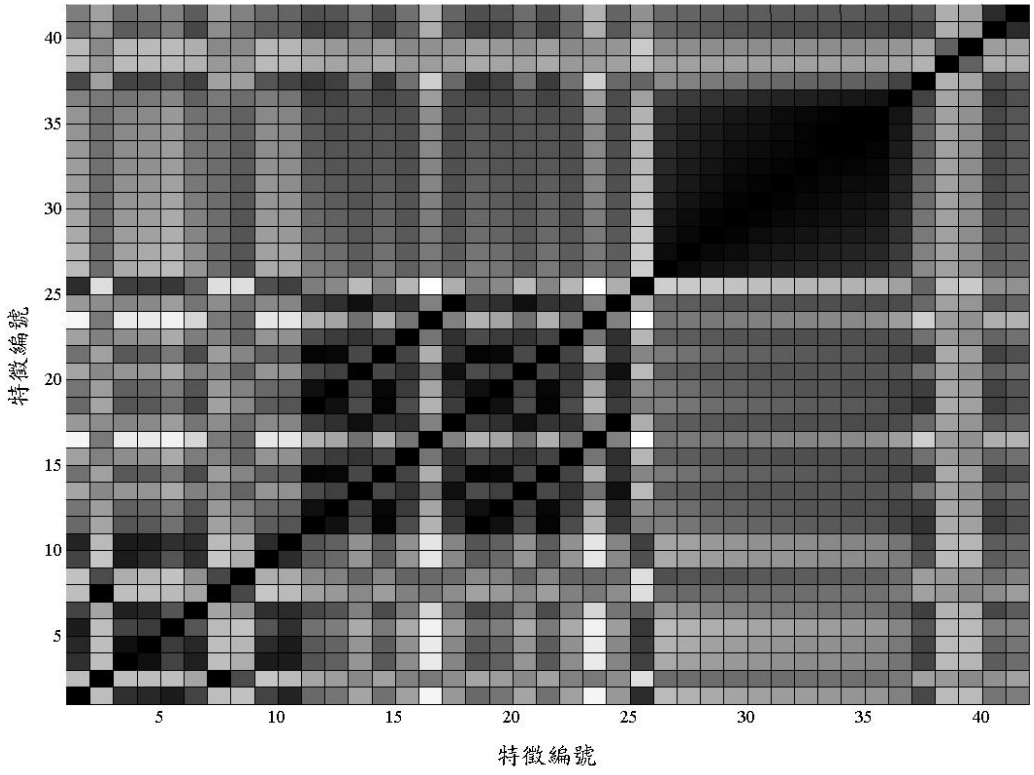

圖 6.15 TDT-2 臺師大大陸口音中文大詞彙語音辨識器轉寫之訓練語料特徵間的 Spearman’s Footrule Distance

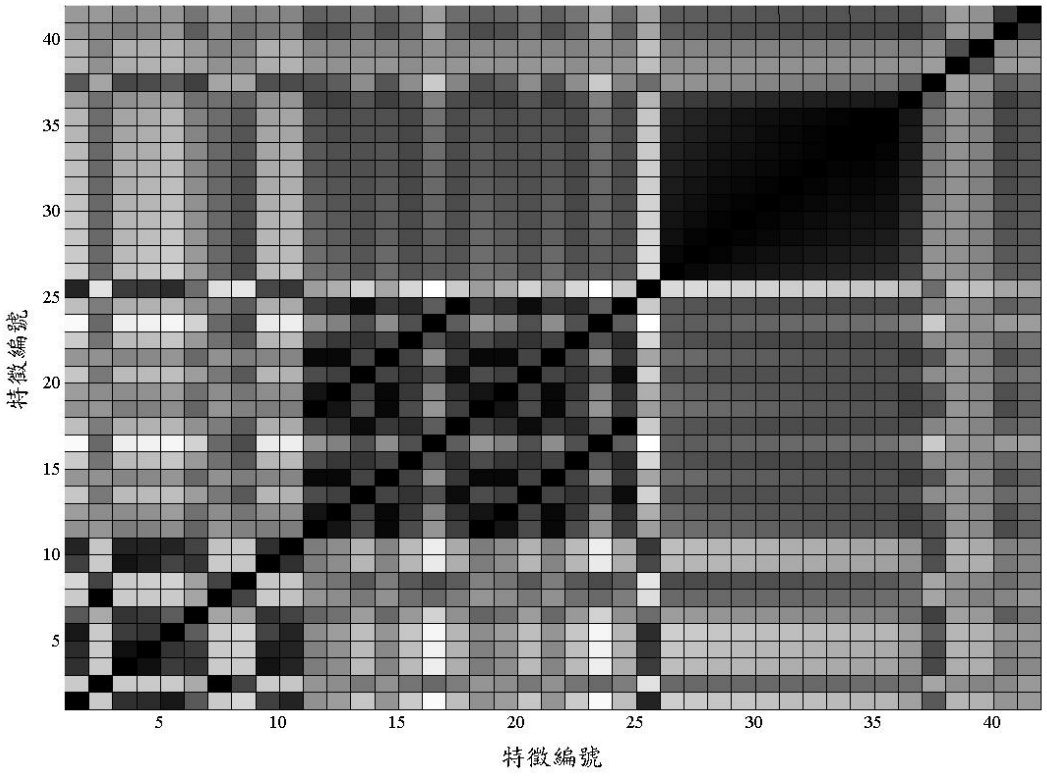

圖 6.16 TDT-3 臺師大大陸口音大詞彙語音辨識器轉寫之訓練語料特徵間的 Spearman’s Footrule Distance

6.16 分別為 TDT-2 及 TDT-3 的訓練語料經過臺師大大陸口音中文大詞彙語音辨識器轉寫的文件，擷取的所有特徵彼此之間的 Spearman’s Footrule Distance。其中，顏色越黑的部份，代表其排序差異大；顏色越淺的部份，代表其排序差異較小。觀察差異性的結果，我們可以得知，TDT-2 及 TDT-3，特徵差異的變化情形類似。

TDT-2	正確率(%)	Precision	實際為相關文件		實際為不相關文件	
			估測正確	估測錯誤	估測正確	估測錯誤
訓練語料	93.12%	0.4192	2067	2864	48442	868
測試語料	98.13%	0.5021	235	233	35329	443

表 6.4 TDT-2 臺師大大陸口音中文大詞彙語音辨識器轉寫之語音文件 SVM 訓練實驗結果分析

TDT-3	正確率(%)	Precision	實際為相關文件		實際為不相關文件	
			估測正確	估測錯誤	估測正確	估測錯誤
訓練語料	94.06%	0.4989	670	673	13226	204
測試語料	99.39%	0.3549	335	609	157143	350

表 6.5 TDT-3 臺師大大陸口音中文大詞彙語音辨識器轉寫之語音文件 SVM 訓練實驗結果分析

觀察了訓練時各項特徵的呈現狀況後，我們再來觀察實際的分類情形，為表 6.4 及表 6.5。由表 6.4 及表 6.5 我們可以發現，TDT-2 經過 SVM 訓練模型之後，測試語料的精確率較高，而 TDT-3 經過 SVM 訓練模型之後，測試語料的精確率卻很低。雖然如此，但 TDT-3 測試語料的正確率卻到達 99%。因此，TDT-3 實驗結果並不理想的原因，可能是訓練的模型，對於不相關文件的估測較為正確，但卻對相關文件的估測不夠準確。然而，一旦相關文件的分類估測不夠精確時，就容易大大的影響到平均精確率的結果。

6.2 成對式訓練在語音文件上的檢索

本節討論排序網路(RankNet)在語音文件上的檢索效能，本論文使用的 RankNet 為兩層的神經網路，即中間有一隱藏層，活化函數(Activation)則選用標準的雙曲正切函數(Hyperbolic Tangent Function)，由於隱藏層的節點數將大大影響檢索效能，故在此實驗中，我們將在訓練語料上將此參數(隱藏層的節點數)調至最佳，即每個節點數(1~100)都會訓練 3 個迭代次數，並比較這 100 個模型在訓練語料上的平均精確率，取最佳的一個，再將此模型拿到測試語料上實驗。

6.2.1 RankNet 在語音正確轉寫上的檢索效能

在本小節中，我們使用成對式訓練中的 RankNet 分別於 TDT-2 與 TDT-3 語料之

語音正確轉寫文件上，觀察其檢索成效。

■ TDT-2

經過實驗，隱藏層的節點數為 2 個時，在 TDT-2 訓練語料上有最佳的平均精確率 0.6013，訓練過程如表 6.6：

迭代次數	平均交互熵	對組錯誤率(%)
1	0.687	12.52
2	0.672	12.38
3	0.649	12.36

表 6.6 RankNet 於 TDT-2 語音正確轉寫迭代過程

由表 6.6 中可見，隨著迭代次數增加，平均交互熵值與對組錯誤率也隨之降低。在 TDT-2 測試語料上的結果如圖 6.17 及圖 6.18。從圖 6.17 中，我們可以發現，RankNet 的平均精確率最高；而在圖 6.18 中，亦可以發現，RankNet 的均化遞減累積獲益在各個位置點，其成效皆最好。因此，在 TDT-2 語音正確轉寫文件上，使用成對式訓練中的 RankNet，能夠有相當不錯的表現。

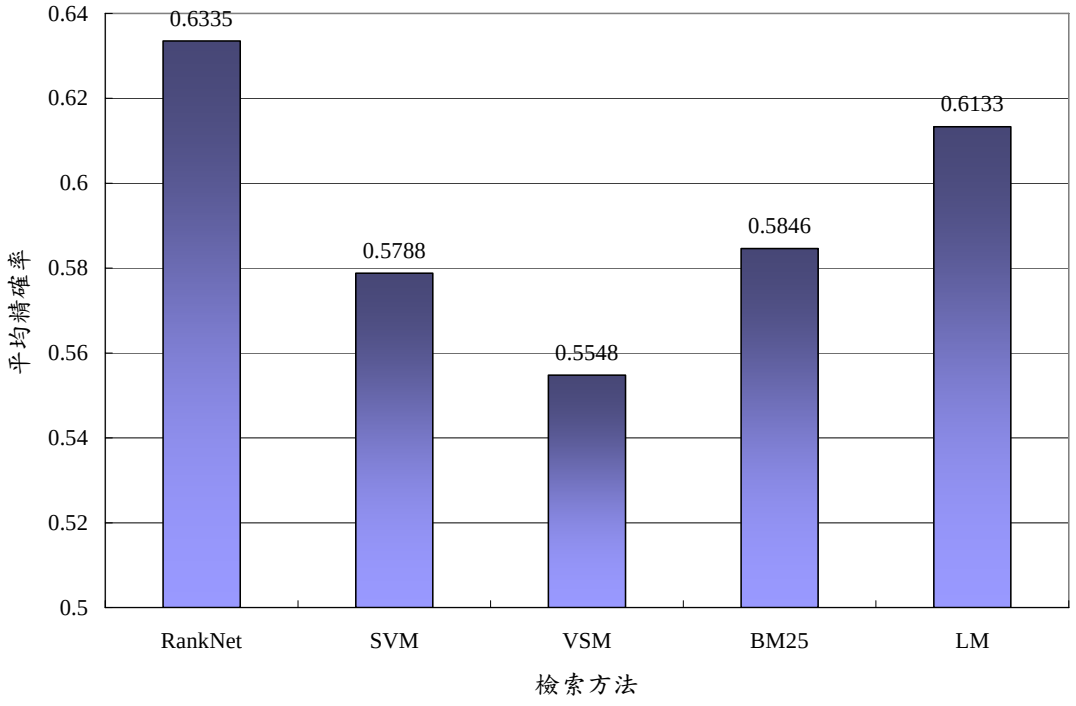

圖 6.17 TDT-2 使用 RankNet 在語音正確轉寫文件之 MAP

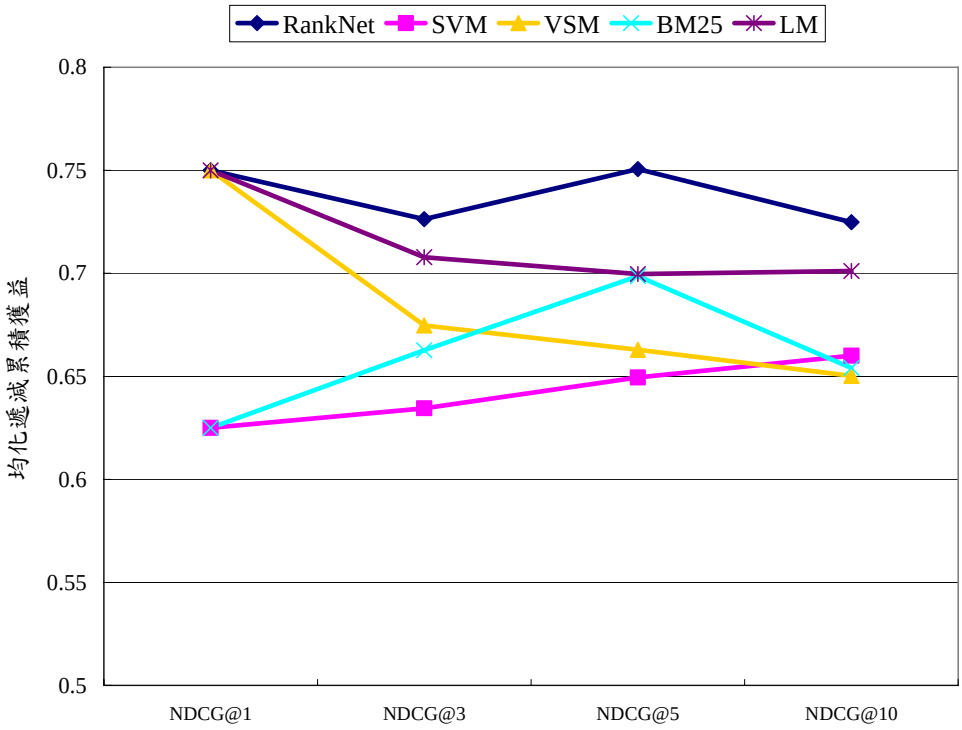

圖 6.18 TDT-2 使用 RankNet 在語音正確轉寫文件之 NDCG

■ TDT-3

經過實驗，隱藏層的節點數為 1 個時，在 TDT-3 訓練語料上有最佳的平均精確率 0.7474，訓練過程如下表：

迭代次數	平均交互熵	對組錯誤率(%)
1	0.689	8.76
2	0.678	8.31
3	0.659	8.00

表 6.7 RankNet 於 TDT-3 語音正確轉寫迭代過程

由表 6.7 中可見，隨著迭代次數增加，平均交互熵值與對組錯誤率也隨之降低。在 TDT-2 測試語料上的結果如圖 6.19 及圖 6.20。在圖 6.19 中，RankNet 的平均精確率一樣為最高；而在圖 6.20 中，RankNet 的均化遞減累積獲益在前 1 名排序狀況雖然不是最佳，但是在前 3 名、前 5 名及前 10 名，其均化遞減累積獲益成

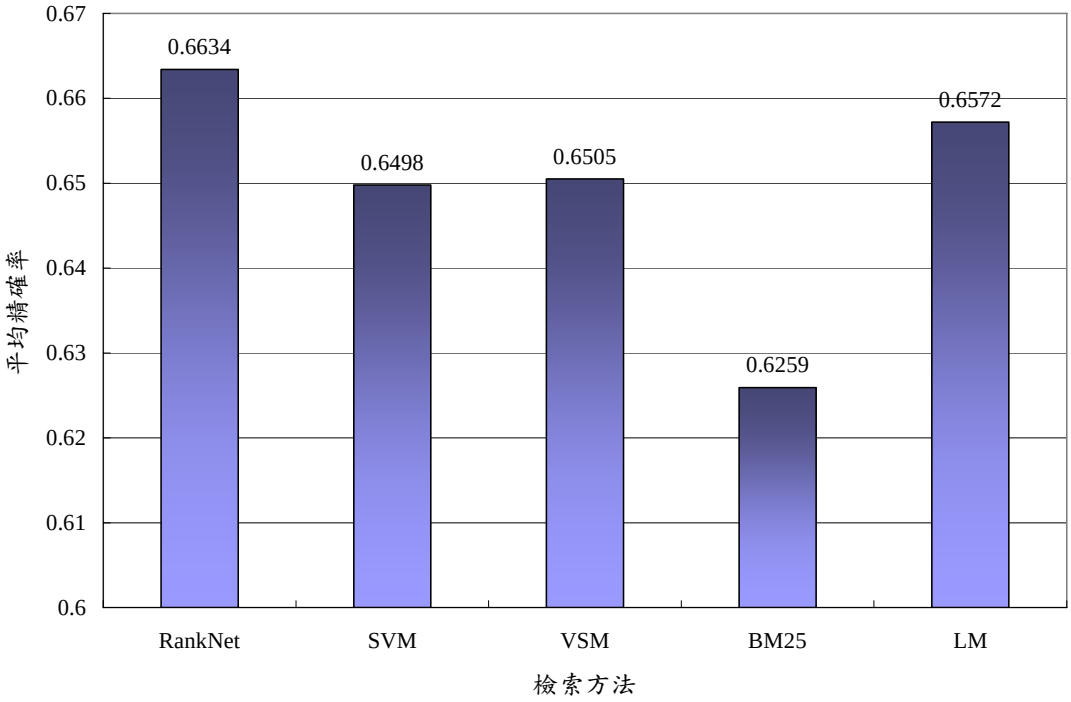

圖 6.19 TDT-3 使用 RankNet 在語音正確轉寫文件之 MAP

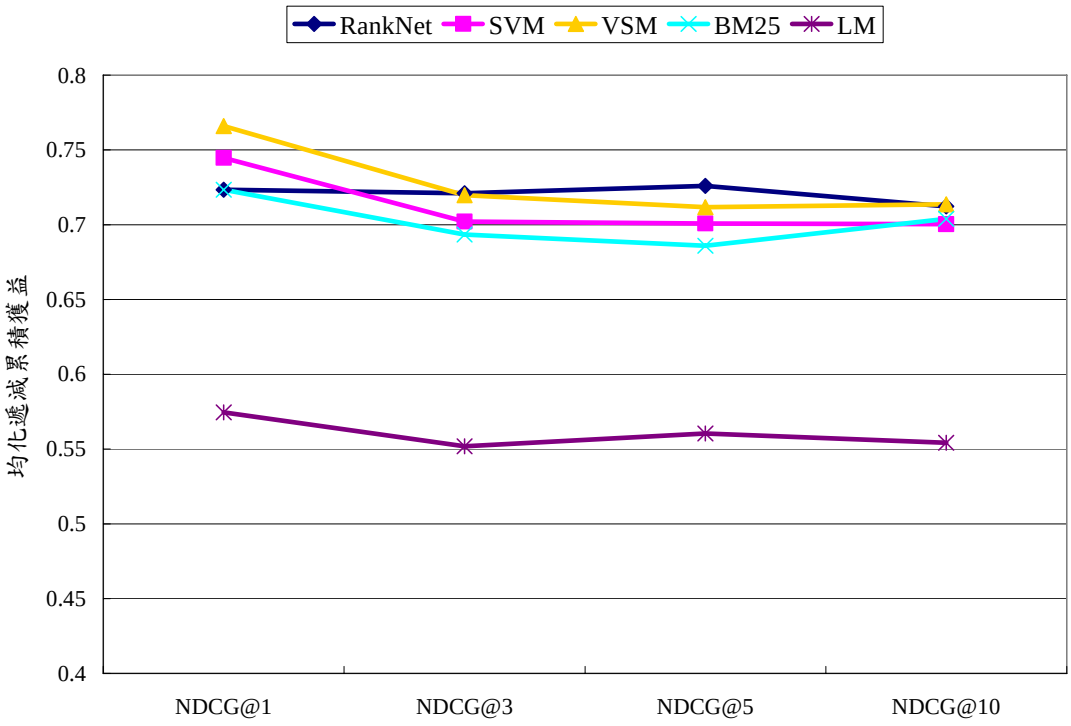

圖 6.20 TDT-3 使用 RankNet 在語音正確轉寫文件之 NDCG

效皆仍然為最好。因此，在 TDT-3 語音正確轉寫文件上，使用成對式訓練中的 RankNet，依然有相當不錯的表現。

由以上實驗，我們可以發現，成對式訓練中的 RankNet 訓練方法在語音正確轉寫之檢索之效果，不論在 TDT-2 或 TDT-3，其平均精確率以及均化遞減累積獲益皆有相當不錯的結果。而此結果，亦較逐點式訓練中的 SVM 訓練為佳。因此，我們認為，資訊檢索的問題，必定不單單只有分類問題，還需要有排序之概念。而 RankNet 相較於 SVM 增加了排序的訓練，才能得到較好的檢索效能。

6.2.2 RankNet 在 Dragon 辨識器轉寫之語音文件的檢索效能

在本小節中，我們將 TDT-2 與 TDT-3 語料經由 Dragon 辨識器轉寫的語音文件，再使用 RankNet 進行訓練，觀察其檢索成效。

- TDT-2

經過實驗，隱藏層的節點數為 3 個時，在訓練語料上有最佳的平均精確率 0.6169，訓練過程如下表：

迭代次數	平均交互熵	對組錯誤率(%)
1	0.683	13.54
2	0.662	13.32
3	0.630	13.27

表 6.8 RankNet 於 TDT-2 Dragon 辨識器轉寫之語音文件迭代過程

由表 6.8 中可見，隨著迭代次數增加，平均交互熵值與對組錯誤率也隨之降低。在 TDT-2 測試語料上的結果如圖 6.21 及圖 6.22。在圖 6.21 中，RankNet 的平均精確率相較於傳統檢索方法以及 SVM 訓練有最高的平均精確率；而在圖 6.22 中，RankNet 的均化遞減累積獲益僅在前 1 名排序較低於 LM，其餘的位置，皆能維持較好的成效。

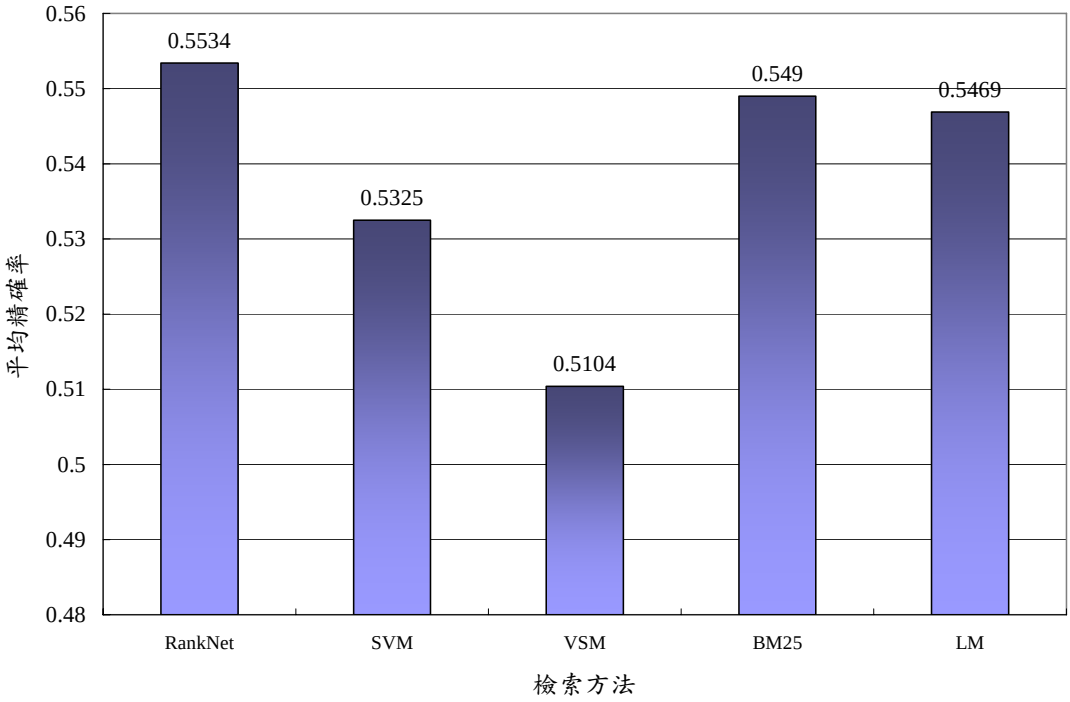

圖 6.21 TDT-2 使用 RankNet 在 Dragon 辨識器轉寫之文件的 MAP

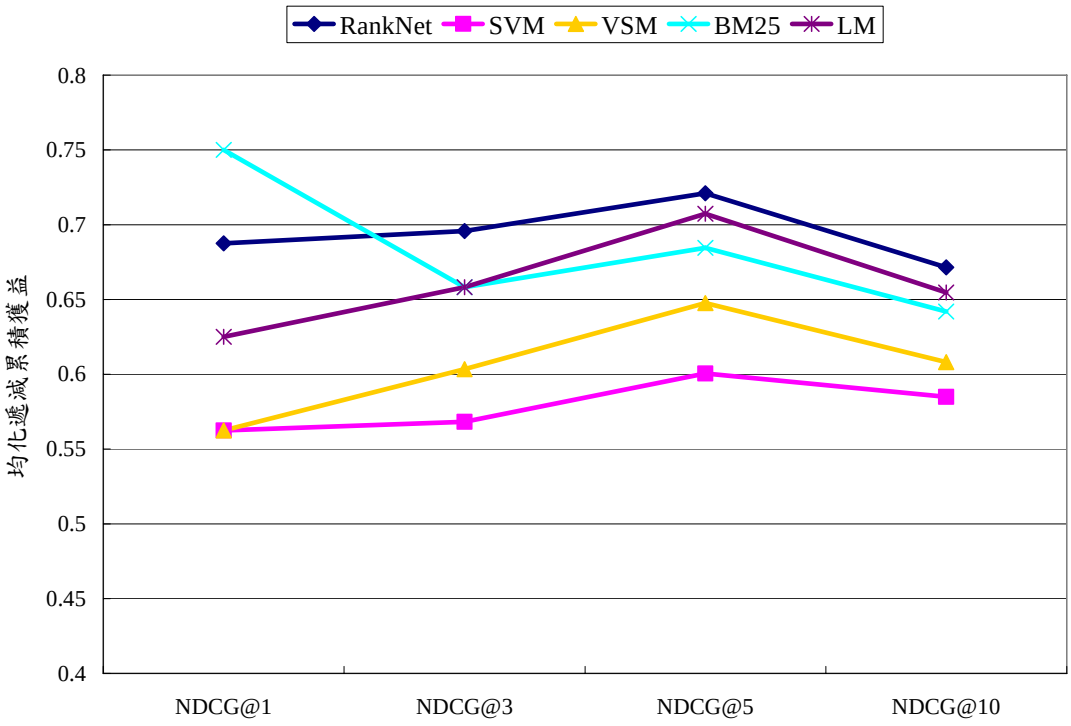

圖 6.22 TDT-2 使用 RankNet 在 Dragon 辨識器轉寫之文件的 NDCG

■ TDT-3

經過實驗，隱藏層的節點數為 1 個時，在訓練語料上有最佳的平均精確率 0.7446，訓練過程如下表：

迭代次數	平均交互熵	對組錯誤率(%)
1	0.689	7.37
2	0.680	6.80
3	0.663	6.45

表 6.9 RankNet 於 TDT-3 Dragon 辨識器轉寫之語音文件迭代過程

由表 6.9 中可見，隨著迭代次數增加，平均交互熵值與對組錯誤率也隨之降低。在 TDT-2 測試語料上的結果如圖 6.23 及圖 6.24。在圖 6.23 中，RankNet 的平均精確率相較於傳統檢索方法以及 SVM 訓練有最高的平均精確率；而在圖 6.24 中，RankNet 的均化遞減累積獲益僅在前 5 位置有最佳的表現。但相較於傳統的檢索

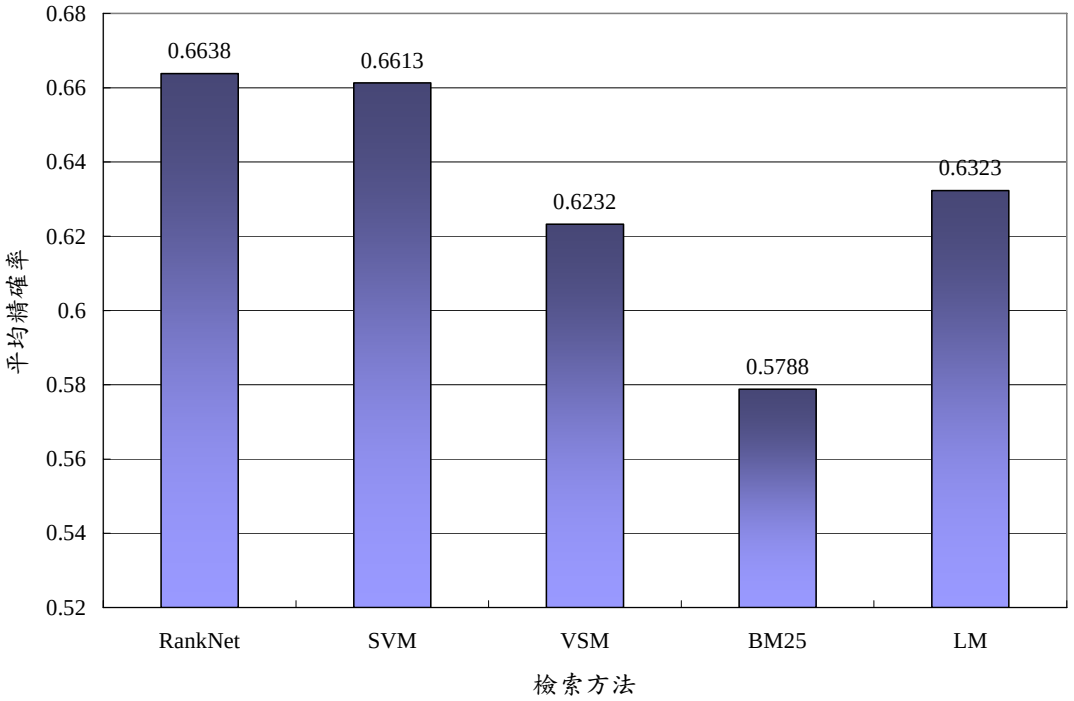

圖 6.23 TDT-3 使用 RankNet 在 Dragon 辨識器轉寫之文件的 MAP

方法，在前 3、前 5 及前 10 位置都較傳統資訊檢索方法成效為佳。此外，我們也發現，RankNet 及 SVM 在 TDT-3 由 Dragon 辨識器轉寫之語音文件，其其訓練後模型之檢索效能，都較 RankNet 及 SVM 於 TDT-3 語音正確轉寫文件之訓練模型為佳。

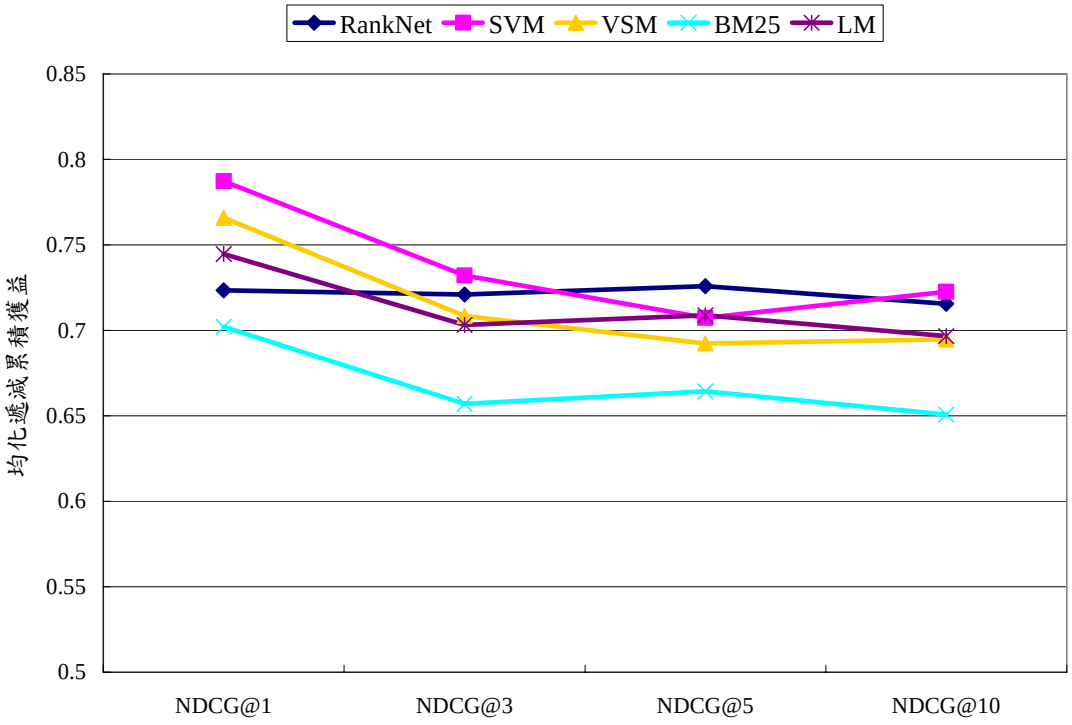

圖 6.24 TDT-3 使用 RankNet 在 Dragon 辨識器轉寫之文件的 NDCG

由以上實驗可以得知，成對式訓練中的 RankNet 訓練方法在使用 Dragon 語音辨識器所轉寫的文件上之檢索效能皆有不錯的成效。因此，RankNet 不僅能幫助語音正確轉寫文件之檢索成效，在有辨識率錯誤之文件中，也能得到提升的效果。

6.2.3 RankNet 在臺師大大陸口音中文大詞彙語音辨識器轉寫之語音文件的檢索效能

在本小節中，我們將 TDT-2 與 TDT-3 語料經由臺師大大陸口音中文大詞彙語音

辨識器所轉寫的語音文件，再使用 RankNet 進行訓練，觀察其檢索成效。

■ TDT-2

經過實驗，隱藏層的節點數為 1 個時，在訓練語料上有最佳的平均精確率 0.5652，訓練過程如下表：

迭代次數	平均交互熵	對組錯誤率(%)
1	0.691	17.38
2	0.687	16.92
3	0.678	16.54

表 6.10 RankNet 於 TDT-2 臺師大大陸口音中文大詞彙語音辨識器轉寫之語音文件迭代過程

由表 6.10 中可見，隨著迭代次數增加，平均交互熵值與對組錯誤率也隨之降低。在 TDT-2 測試語料上的結果如圖 6.25 及圖 6.26。在圖 6.25 中，RankNet 的平均精確率較 SVM 為差，但較其它的傳統資訊檢索方法為佳。在圖 6.26 中，RankNet

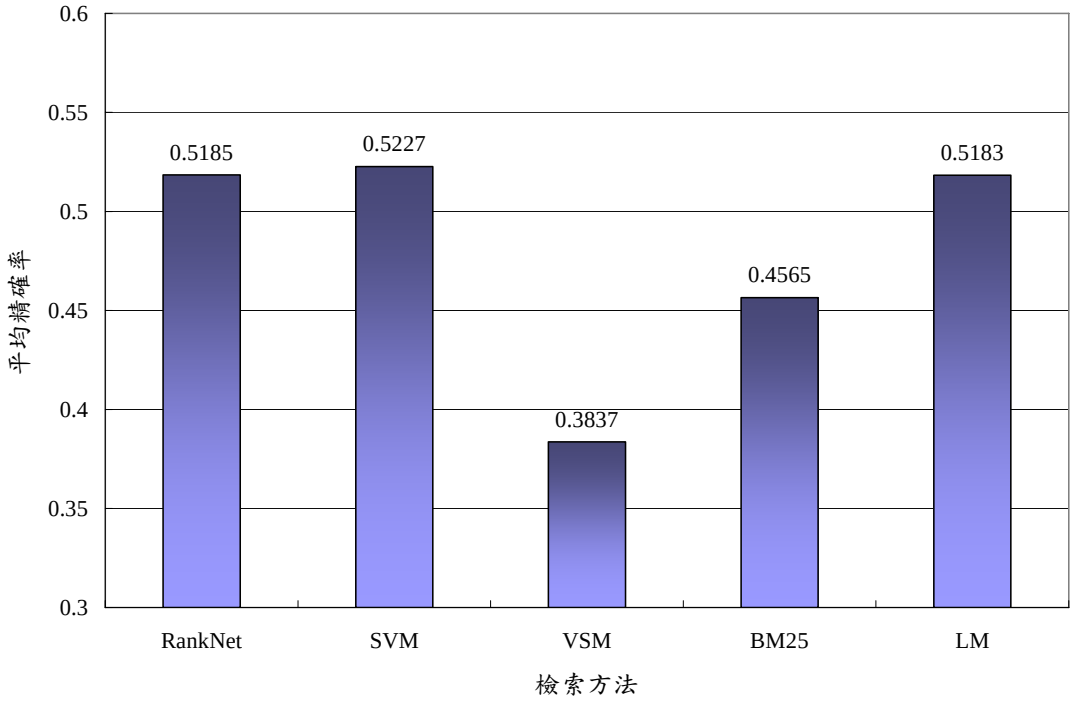

圖 6.25 TDT-2 使用 RankNet 在臺師大大陸口音中文大詞彙語音辨識器轉寫之語音文件的 MAP

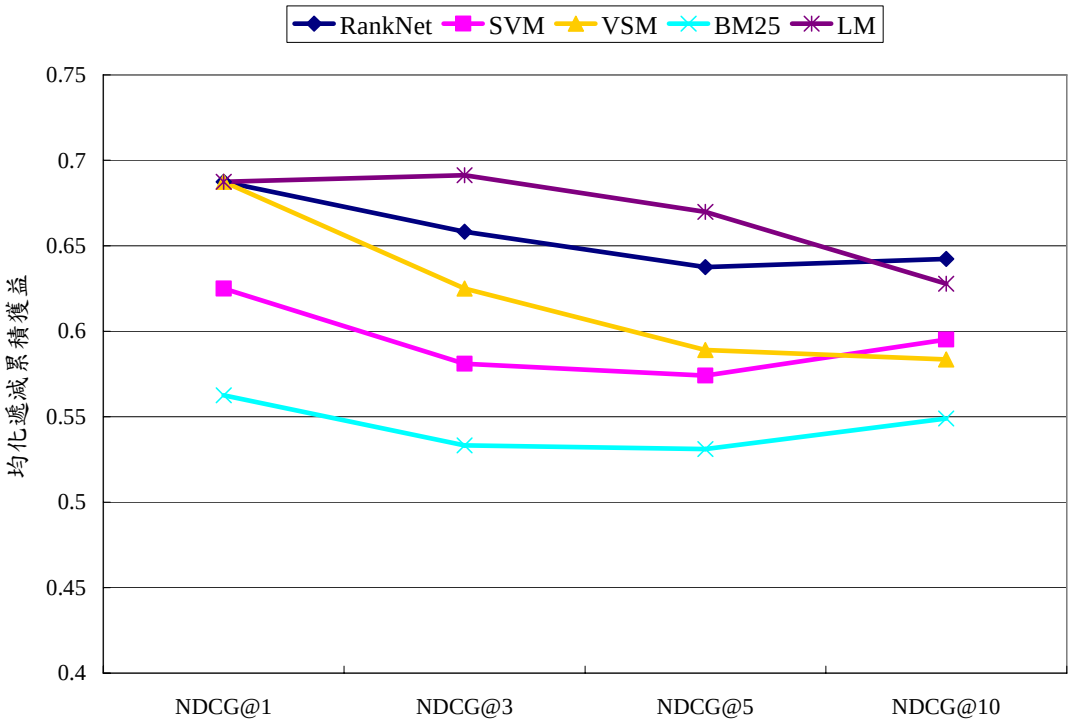

圖 6.26 TDT-2 使用 RankNet 在臺師大大陸口音中文大詞彙語音辨識器轉寫之語音文件的 NDCG

的均化遞減累積獲益的表現在前 1、前 3 及前 5 位置時，較 LM 為差，但在前 10 位置時，有最佳的效能。

■ TDT-3

經過實驗，隱藏層的節點數為 1 個時，在訓練語料上有最佳的平均精確率 0.6483，訓練過程如下表：

迭代次數	平均交互熵	對組錯誤率(%)
1	0.690	11.62
2	0.683	10.69
3	0.670	10.05

表 6.11 RankNet 於 TDT-3 臺師大大陸口音中文大詞彙語音辨識器轉寫之語音文件迭代過程

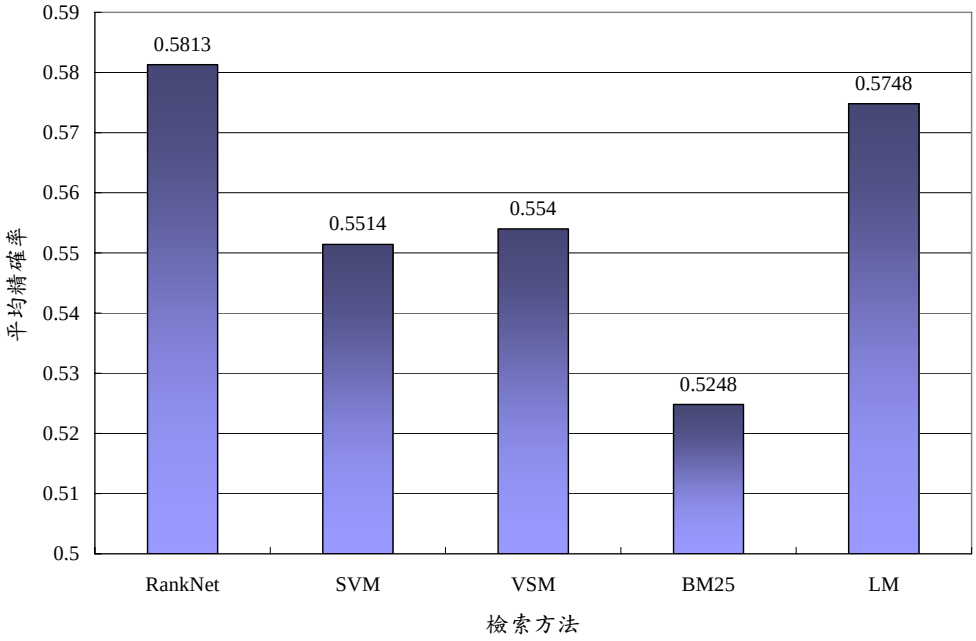

圖 6.27 TDT-3 使用 RankNet 在臺師大大陸口音中文大詞彙語音辨識器轉寫之語音文件的 MAP

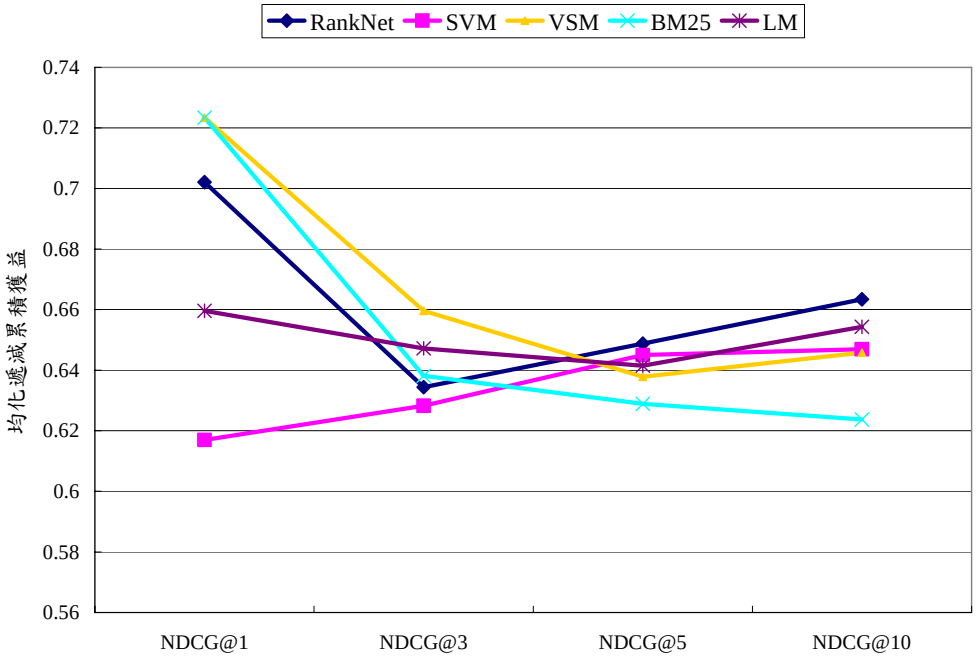

圖 6.28 TDT-3 使用 RankNet 在臺師大大陸口音中文大詞彙語音辨識器轉寫之語音文件的 NDCG

由表 6.11 中可見，隨著迭代次數增加，平均交互熵值與對組錯誤率也隨之降低。在 TDT-2 測試語料上的結果如圖 6.27 及圖 6.28。在圖 6.27 中，RankNet 的平均精確率成效最好。而圖 6.28 中，RankNet 在均化遞減累積獲益之成效上，在前 1 位置及前 3 位置較不理想，但是在前 5 位置及前 10 位置都有最佳的表現。

由以上實驗結果，我們可以得知，在較低辨識率的語音文件上，亦即，含有較多錯誤的語音文件，RankNet 依然能較傳統資訊檢索方法有較好的檢索成效。

6.3 成對式訓練與平均精確率之關係

在 6.2 章中，我們討論了成對式訓練方法，皆有不錯的成效。然而，在進行實驗時，我們觀察到了此種訓練方法並不完全能夠提升平均精確率。首先，倘若我們

文件編號	1	2	3	4	5	6	7	8
正確相關度 相關=1 不相關=0	1	0	0	0	0	0	0	1

(a)

文件編號	1	2	3	4	5	6	7	8
正確相關度 相關=1 不相關=0	0	0	1	1	0	0	0	0

(b)

	對組正確率	平均精確率
訓練前	6	0.6250
訓練後	8	0.4167

(c)

表 6.12 成對式訓練與平均精確率比較範例

對表 6.12 (a)的資料進行訓練。而成對式訓練的目標在於增加對組正確率。在表 6.12 (a)中，有兩筆相關文件，對文件編號 1，有 6 個正確對組；對文件編號 8，有 6 個錯誤對組。因此，表 6.12 (a)的總體正確對組共有 6 個。當我們訓練後的結果為表 6.12 (b)時，對文件編號 3 及文件編號 4，各有 4 個正確對組及 2 個錯誤對組。因此，表 6.12 (b)較表 6.12 (a)增加了對組的正確率。然而，在平均精確率上，訓練前為 0.6250，而訓練後為 0.4167，經過訓練後，平均精確率反而下降了。

這樣的狀況通常發生在抽取特徵，若特徵的排序狀況，相關文件分布相當極端時。而這種情況其實經常出現。訓練的目的，可能使得相關文件的分布較不極端，但這樣的結果不見得會提高平均精確率。

6.4 使用更新方法解決不平衡語料問題之實驗

在本節中，我們將本論文所提出之更新方法使用於 TDT-2 語料庫，並使用在經

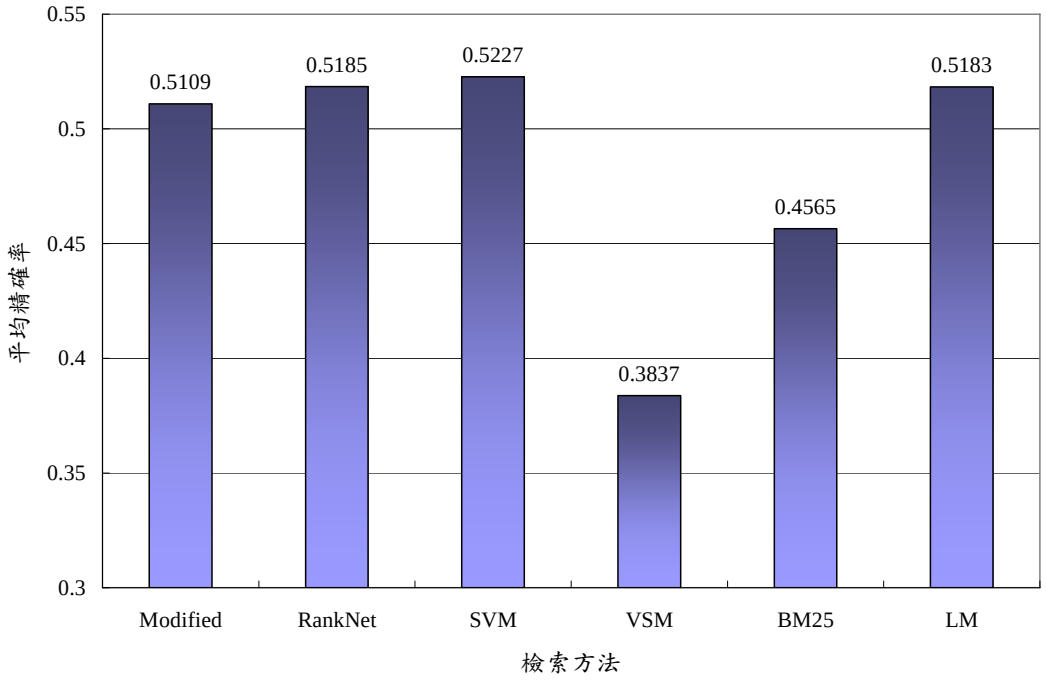

圖 6.29 TDT-2 臺師大大陸口音中文大詞彙語音辨識器轉寫之語音文件不平衡語料問題更新方法之 MAP

過臺師大大陸口音中文大詞彙語音辨識器所轉寫的語音文件中。對訓練語料進行處理後，再使用成對式訓練中的 RankNet 進行訓練。實驗結果為圖 6.29 及圖 6.30 所示。圖 6.29 使用了更新方法之後的平均精確率與未經過處理以及傳統資訊檢索方法進行比較。

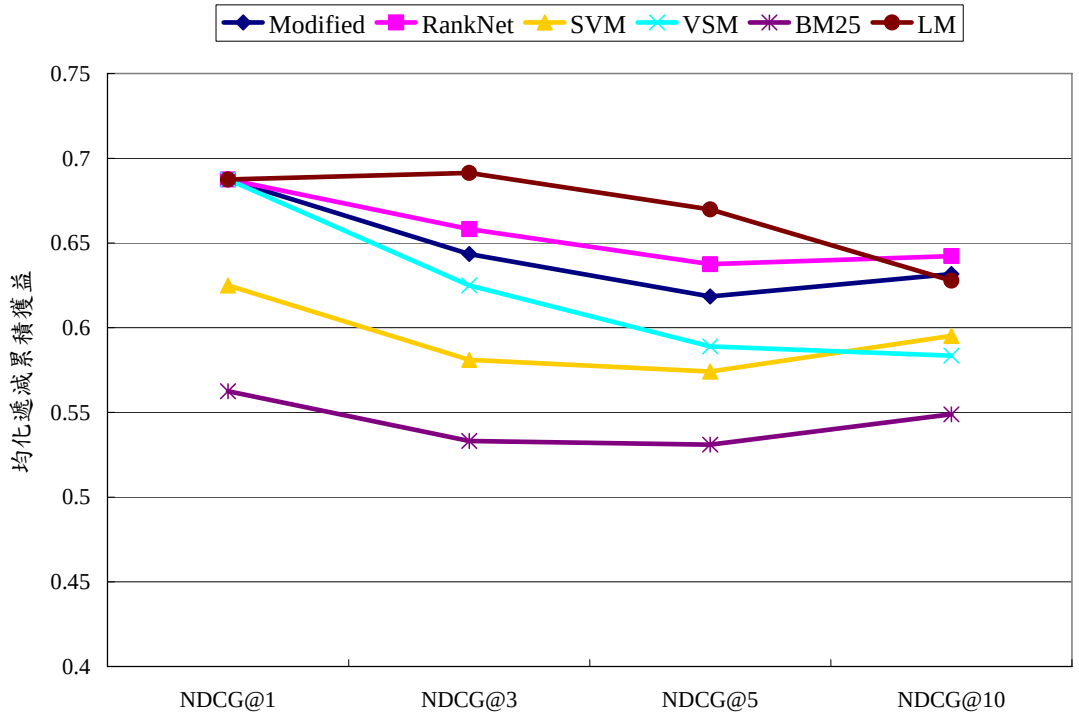

圖 6.30 TDT-2 臺師大大陸口音中文大詞彙語音辨識器轉寫之語音文件不平衡語料問題更新方法之 NDCG

由圖 6.29 可以得知，經過改變的訓練語料，訓練模型後，並無法得到比原先的訓練語料進行訓練的模型，在平均精確率的表現更好。在圖 6.30 中，一樣可以發現，經過改變的訓練語料，其所訓練出來的模型在均化遞減累積獲益上也不能有所提升。

在這樣的結果下，我們檢視訓練語料的分群是否有誤。由於，我們對所有的文件以及訓練查詢一起進行分群，所以，我們可以得到所有文件的分群狀況。而我們又已經擁有所有的測試查詢對應到所有文件為相關的正確解答。因此，若每一個測試查詢對應的相關文件，在我們分群時都被分在同一群之中，我們認為這

樣的分群是沒有問題的。觀察分群的結果如表 6.12。我們可以看到，僅有 20002.query 和 20048.query 中的對應相關文件在分群時有被分到兩大群中，20002.query 相關文件數共有 13 篇，13 篇中有 12 篇被分到第 2 群；有 1 篇被分到第 3 群。而其它，如 20001.query 相關文件數共有 15 篇，這 15 篇群全部都被分到第 14 群。因此，我們可以得知，上述實驗結果，在訓練語料的分群時，並沒有錯誤。

測試查詢	相關文件數 (篇)	分群結果(文件數→群編號)
20001.query	15	15 →14
20002.query	13	12 → 2； 1 → 3
20005.query	14	14 → 10
20013.query	5	5 → 8
20015.query	87	87 → 8
20020.query	3	3 → 16
20023.query	2	2 → 4
20039.query	35	35 → 14
20048.query	3	2 → 4；1 → 19
20070.query	84	84 → 14
20071.query	31	31 → 15
20076.query	70	70→ 14
20088.query	2	2 → 7
20089.query	13	13 → 17
20091.query	9	9 → 6
20096.query	4	4 → 0

表 6.12 TDT-2 臺師大大陸口音中文大詞彙語音辨識器轉寫之語音文件不平衡語料問題更新方法之分群測試

改變訓練語料之不平衡狀況，目前仍無法對檢索結果進行改善，其原因可能是訓練語料過於混淆，我們擷選出的正例增加過多，造成雜訊過多，影響了訓練結果。此外，正例與反例的比例是相當難以拿捏的，而其對訓練結果亦有很大的

影響。對於正反例的選取，不僅是本論文進行討論，在微軟團隊於 2008 年公開發表的 LETOR 3.0 版(此為提供一套純文字的資訊檢索語料，並針對排序學習於資訊檢索議題上的各種討論)中，在說明文件的草稿中，他們也同樣針對訓練語料擷選的問題進行討論。最剛開始，他們使用 BM25 對文件進行排序，先選出所有標示為正例者的資料點，接著選出經由排序後前 n 筆反例的資料點。然而，資訊檢索大師 Rijsbergen 認為，不應該先選出所有標示為正例者，應該直接選擇經由 BM25 排序後前 m 筆資料點做為訓練資料點。其理由為，當有一正例資料點，其 BM25 分數很低，但是卻擷選為訓練資料點，那麼就會存在一種現象：訓練資料中含有 BM25 分數高者亦含有 BM25 分數低者。這樣的資料在訓練時就會產生混淆，不知道訓練模型該符合 BM25 分數高者，還是該符合 BM25 分數低者。

因此，即便我們對訓練語料進行的改變並不能達到改善的效果，但訓練語料的問題，仍然是大家著重的議題。

7. 結論

本論文引入排序學習之逐點式訓練的 SVM 及成對式訓練的 RankNet 進行訓練。並將排序學習方法應用於語音文件的資訊檢索上。我們發現，RankNet 在語音文件資訊檢索上，較傳統資訊檢索方法為佳。並且相較於 SVM，RankNet 於語音正確轉寫文件、Dragon 語音辨識器轉寫的語音文件及臺師大大陸口音中文大詞彙語音辨識器轉寫的語音文件，資訊檢索上的成效皆較佳。這可以說明，資訊檢索的問題並不單單只是分類概念，而是具有排序問題。因此，若要得到更佳之檢索成效，我們必須朝向以排序做為最佳化之評估方式之方向，選用訓練模型。

傳統資訊檢索方法，很容易因為文件隱含有錯誤資訊，而造成檢索效能的大幅降低，其檢索效能並不穩定。並且，我們也不能確知參數的設定該如何訂定最能符合語料需求。而透過排序學習的方式，我們可以對具有錯誤資訊的語音文件進行訓練，並納入多種傳統資訊檢索的方法。

在本論文中，我們仔細的分析了 SVM 運用於資訊檢索上的問題，也觀察到成對式訓練潛在的問題，在本論文中的成對式訓練其實有不錯的成效，但我們依然發現，成對式訓練並不能完全確保其訓練的結果能與平均精確率有正相關的關係。

此外，在資訊檢索上使用機器學習方法進行訓練，必須考量到資訊檢索問題上特有的不平衡問題，雖然在本論文中，我們尚未能夠顯著提升檢索效能，但是，利用有意義的資料選取方法，必定是資訊檢索使用機器學習方法中勢必考量的。

8. 未來展望

將檢索問題考慮為一種排序問題，並進行訓練時，可以進行的研究必定相當之多。在第三章中，我們提到的特徵擷取問題，亦是需要探討之課題。在我們大量對排序學習議題於資訊檢索上之研究了解之後，並沒有發現針對何種特徵擷取對於資訊檢索最具成效之探討，僅有[Geng et al. 2007]對大量特徵進行篩選。然而，選擇幫助較大之特徵才進行擷取，不僅便利了訓練資料的建立，更能增進訓練時間之速度。因此，特徵的擷取問題仍是一個極具有探討空間的議題。尤其，語音文件是一種特殊的文件型態，納入語音的低階特徵，讓可見的資訊更多亦是一種特徵擷取的方法。語音文件若能經由機器學習方法達到較佳的檢索成效，那麼選擇了語音特有之資訊作為特徵，想必能夠得到更好的檢索成果。

2009 年，[Chien & Wu 2009]對語言模型(Language Modeling)改進了以鑑別式訓練(Discriminative Training)概念發展出之最小分類錯誤(Minnimum Classification Error, MCE) [Chen et al. 2004a]的方法，提出最小排序錯誤(Minimum Rank Error, MRE)，此改善的方法也可視為一種排序學習。因此，此種鑑別式訓練方法未來可與 SVM 或 RankNet 等作深入的比較。

除此之外，資訊檢索的語料問題仍就是一個開放的議題。如何使用較具意義的方法來選擇訓練語料，使得訓練語料更具有代表性。如果能有效的選擇訓練語料，就能大幅減少訓練的時間，並且能夠達到較好的訓練結果。

9. 參考文獻

[Baeza-Yates & Ribeiro-Neto 1999] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*, 1999.

[Bai et al. 2000] B. R. Bai, B. Chen, H.-M. Wang. Syllable-based Chinese text/spoken document retrieval using text/speech Queries. *International Journal of Pattern Recognition and Artificial Intelligenc*e, 14(5), pp. 603-616, August 2000.

[Bashir et al. 2002] Faisal I. Bashir, Ashfaq A. Khokhar. Video Content Modeling: An Overview**.** Technical Report, Department of CS/ECE, UIC, 2003.

[Berger 2001] Berger, A. Statistical Machine Learning for Information Retrieval. Doctoral Thesis, Carnegie Mellon University.

[Boser et al. 1992] B.E. Boser, I.M. Guyon, and V.N. Vapnik. A training algorithm for optimal margin classifier. In *Proc. 5th ACM Workshop on Computational Learning Theory*, pp. 144-152, Pittsburgh, PA, July 1992.

[Brin & Page 1998] S. Brin, and L. Page. The Anatomy of a Large-Scale Hypertextual Web Search Engine. In *Seventh International World-Wide Web Conference (WWW)*, April 14-18, Brisbane, Australia, 1998.

[Burges et al. 2005] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to rank using gradient descent. In *ICML '05: Proceedings of the 22nd international conference on Machine learning*, pp. 89-96, New York, NY, USA, 2005.

[Burges et al. 2007] Christopher J.C. Burges, Robert Ragno and Quoc Viet Le. Learning to Rank with Nonsmooth Cost Functions. In *Advances in Neural Information Processing Systems: Proceedings of the 2006 Conference*, MIT Press, 2007.

[Cao et al. 2007] Z. Cao, T. Qin, T.-Y. Liu, M.-F. Tsai, and H. Li. Learning to Rank: From pairwise approach to listwise approach. In Z. Cao, T. Qin, T.-Y. Liu, M.-F. Tsai, and H. Li. Learning to rank: from pairwise approach to listwise approach. In *ICML '07: Proceedings of the 24th international conference on Machine learning*,

pp. 129-136, New York, NY, USA, 2007.

[Carbonell & Goldstein 1998] Jaime G. Carbonell, Jade Goldstein: The Use of MMR, Diversity-Based Reranking for Reordering Documents and Producing Summaries. In *Proc. SIGIR'98*, pp. 335-336, 1998

[Chang 1997] Shih-Fu Chang. Content-Based Indexing and retrieval of visual information. *IEEE Signal Processing Magazine*, 14(4), pp. 45-48, July 1997.

[Chang & Lin 2001] Chih-Chung Chang and Chih-Jen Lin, LIBSVM : a library for support vector machines, 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm

[Chor et al. 1998] B. Chor, O. GoldReich, E. Kushilevitz. and M. Sudan. Private Information Retrieval. *Journal of the ACM*, Vol. 45, No. 6, pp. 965–982. 1998.

[Chen et al. 2004a] B. Chen, H.-M. Wang, and L.-S. Lee, A discriminative HMM/N-gram-based retrieval approach for Mandarin spoken documents. *ACM Transactions on Asian Language Information Processing*, Vol. 3, No. 2, pp. 128-145, June 2004.

[Chen et al. 2004b] B. Chen, J.-W. Kuo, and W.-H. Tsai, Lightly supervised and data-driven approaches to mandarin broadcast news transcription. *In Proc ICASSP*, 2004.

[Chen et al. 2005] B. Chen, J.-W. Kuo, W.-H. Tsai, Lightly supervised and data-driven approaches to mandarin broadcast news transcription. *Internation Journal of Computational Linguistics & Chinese Language Processing,*, Vol. 10, No. 1, pp 1-18, 2005.

[Chen 2006] B. Chen. Exploring the use of latent topical information for statistical Chinese spoken document retrieval. *Pattern Recognition Letters*, Vol. 27, Issue 1, pp. 9-18, January 2006.

[Chen 2006] B. Chen. Voice retrieval of Mandarin broadcast news speech. *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 20, No. 1, pp. 91-109, February 2006.

[Chien & Wu 2009] J.-T. Chien & M.-S. Wu. Minimum rank error language modeling. *IEEE TASL*. Vol. 17, No. 2, 2009.

[Clements et al. 2002] M. Clements, S. Robertson, and M. Miller. Phonetic searching applied to on-line distance learning modules. In *Proceedings of 2002 IEEE 10t Digital Signal Processing Workshop, 2002, and the 2nd Signal Processing Education Workshop*, pp. 186–191, 2002.

[Cortes & Vapnik 1995] C. Cortes and V. Vapnik. Support Vector Networks. *Machine Learning*, 20, pp. 1-25, 1995.

[Diaconis 1998] P. Diaconis. *Group Representation in probability and statistics.* In *IMS Lecture Series*, No. 11, Institute of Methmatical Statistics, 1988.

[Drucker et al. 1999] H. Drucker, D. Wu, and V. N. Vapnik. Support vector machines for spam categorization. *IEEE Transactions on Neural Networks*, 10, pp. 1048-1054, 1999.

[Flickner et al. 1995] Myron Flickner, Harpreet Sawhney, Wayne Niblack, Jonathan Ashley, Qian Huang, Byron Dom, Monika Gorkani, Jim Hafner, Denis Lee, Dragutin Petkovic, David Steele, and Peter Yanker. Query by image and video content: the QBIC system. *IEEE Computer* 28(9), pp. 23-32, 1995.

[Frey & Dueck 2007] B. J. Frey, and D. Dueck. Clustering by passing messages between data points. *Science*, 315, pp. 972-976, 2007.

[Furnas et al. 1988] G. W. Furnas, S. Deerwester, S. T. Dumais, T. K. Landauer, R. A. Harshman, L. A. Streeter, and K. E. Lochbaum. Information retrieval using a singular value decomposition model of latent semantic structure. In *SIGIR ’88*: *Proceeding of the 11th annual international ACM SIGIR conference in Research and development in information retrieval*, pp. 465-480, 1988.

[Garofolo et al. 2000] J. Garofolo, G. Auzanne, and E. Voorhees. The TREC spoken document retrieval track: A success story. In *Proceedings of the Ninth Text Retrieval Conference (TREC-9)*, National Institute of Standards and Technology (NIST), 2000.

[Geng et al. 2007] X. Geng, T.-Y. Liu, T. Qin, H. Li. Feature Selection for Ranking. In

Proceedings of the 30nd annual international ACM SIGIR conference on Research and development in information retrieval, pp. 407-414, 2007.

[Geng et al. 2008] X. Geng, T.-Y. Liu, T. Qin, A. Arnold, and H. Li, H.-Y. Shum. Query dependent ranking using K-Nearest neighbor. In *Proc. SIGIR '08*, pp. 115-122, 2008.

[Goodrum 2000] Abby A. Goodrum. Image Information Retrieval: An Overview of Current Research. *Informing Science*, Vol. 3, No. 2, 2000.

[Hardy et al. 2002] H. Hardy, N. Shimizu, T. Strzalkowski, L. Ting, X. Zhang, G.. Bowden Wise. Cross-document summarization by concept classification. In *SIGIR '02*: *Proceeding of the 29th annual international ACM SIGIR conference in Research and development in information retrieval*, pp. 121-128, 2002.

[Herbrich et al. 2000] R. Herbrich, T. Graepel, and K. Obermayer. Large margin rank boundaries for ordinal regression. *Advances in Large Margin Classifiers*, pp. 115-132, 2000.

[Hsu et al. 2008] C.-W. Hsu, C.-C. Chang, C.-J. Lin. A practical guide to support vector classification. Technical report, Department of Computer Science and Information Engineering, National Taiwan University, Taipei, 2003. http://www.csie.ntu.edu.tw/↑cjlin/libsvm/. 2008.

[Järvelin & Kekäläinen 2002] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Transaction on Information Systems*, 20(4), pp. 422-446, 2002.

[Joachims 1997] T. Joachims. A probabilistic analysis of the Rocchio Algorithm with TFIDF for categorization. In *Proc*. *ICML'97*, 1997.

[Joachims 1998] T. Joachims. Text categorization with support vector machines: learning with features. In *Proceedings of European Conference on Machine Learning*, pp. 137-142, 1998.

[Kendall & Gibbons 1990] M. Kendall and J. D. Gibbons. *Rank Correlation Methods*. Edward Arnold, London, 1990.

[Keselj 1997] V. Keselj. Natural language parsing for internet information retrieval. In *Proceedings of 1997 TRIO/ITRC Researcher Retreat*, 1997.

[Kleinberg 1999] J. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5), pp. 604-632, 1999.

[Li 2007] P. Li, C.J.C. Burges, and Q Wu. McRank:Learning to rank using multiple classification and gradient boosting. *In NIPS2007*, 2007.

[Liu et al. 2007] T.-Y. Liu, J. Xu, T. Qin, W. Xiong, and H. Li. LETOR:Benchmark dataset for research on learning to rank for information retrieval. In *Proc. SIGIR 2007 Workshop on Learning to Rank for Information Retrieval.*

[Liu 2008] T.-Y. Liu. Learning to rank for information retrieval. Tutorial at *17th International World-Wide Web Conference (WWW)*, 2008.

[Luenberger 1984] D. G. Luenberger. *Linear and Nonlinear Programming*. Addison-Wesley, 1984.

[Luhn 1958] H.P. Luhn. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, pp. 159-165, 1958.

[MacQueen 1967] J. B. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability*, Berkeley, University of California Press, 1, pp. 281-297. 1967.

[Mamou et al. 2006] J. Mamou, D. Carmel, and R. Hoory. Spoken document retrieval from call-center conversations. In *SIGIR '06*: *Proceeding of the 29th annual international ACM SIGIR conference in Research and development in information retrieval*, pp. 51-58, 2006.

[Manning et al. 2007] C. D. Manning, P. Raghavan, H. Schütze. *An Introduction to Information Retrieval*. Cambridge University Press, Cambridge, England, 2007.

[Masand et al. 1992] B. Masand, G. Linoff, and D. Waltz. Classifying news stories using memory based reasoning. In *Proc. SIGIR '92*, pp. 59-65, 1992.

[Meng et al. 2007] S. Meng, P. Yu, F. Seide, and J. Liu. A Study of Lattice- Based Spoken Term Detection. In *Proc. ASRU '07*, 2007.

[Meteer et al. 1991] M. Meteer, R. Schwartz, and R. Weischedel.. Studies in part of speech labeling. In *Proceedings of the 4th DARPA Speech and Natural Language Workshop* (pp. 331-336). San Mateo, CA: Morgan-Kaufmann, 1991.

[Miller et al. 1999] David R. H. Miller, Tim Leek, Richard M. Schwartz. A Hidden Markov Model Information Retrieval System. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 214-221, 1999.

[Moffat & Zobel 2008] A. Moffat, and J. Zobel. Rank-biased precision for measurement of retrieval effectiveness. *ACM Transactions on Information Systems*, Vol.27, No. 1, 2008.

[Nallapati 2004] R. Nallapati. Discriminant models for information retrieval. In *SIGIR '04*: *Proceeding of the 27th annual international ACM SIGIR conference in Research and development in information retrieval*, pp. 64-71, UK, 2004.

[NIST 2006] The Spoken Term Detection (STD) 2006 Evaluation Plan. NIST, 2006.

[Ortmanns et al. 1997] S. Ortmanns, H. Ney, X. Aubert, A word graph algorithm for Large Vocabulary continuous speech recognition. Computer Speech & Language. Pp. 43-72. 1997.

[Otsuji et al. 1991] K. Otsuji, Y. Tonomura, and Y. Ohba. Video browsing using brightness data. In *Proc. SPIE/IS&T VCIP'91*, Vol. 1606, pp. 980–989, 1991.

[Petkovic et al. 2002] M. Petkovic, R. Zwol, H. E. Blok, W. Jonker, P. M. G. Apers, M. Windhouwer, M. Kersten. Content-based Video Indexing for the Support of Digital Library Search. In *Proc. 18th IEEE International Conference on Data Engineering (ICDE)*, San Jose, USA, February 2002.

[Rijsbergen 1979] C. J. van Rijsbergen. *Information retrieval*. Butterworths, 1979.

[Roberson et al. 1976] S. E. Roberson and K. Sparck Jones. Relevance weighting of search terms. *Journal of American Society for Information Sciences*, 27(3), pp.

129-146, 1976.

[Roberson et al. 1995] S. E. Roberson, S. Walker, S. Jones, M. M. IIancock-Beaulieu, and M. Gatford. Okapi at TREC-3. In *Proceedings of the Third Text Retrieval Conference (TREC)*, 1995.

[Rocchio 1971] J. J. Rocchio. Relevance feedback in information retrieval. *The SMART Retrieval System: Experiments in Automatic Document Processing*, pp. 313-323, Englewood Cliffs, NJ: Prentice-Hall, 1971.

[Salton 1968] G. Salton, and M. E. Lesk. Computer evaluation of indexing and text processing. *Journal of the Association for Computing Machinery*, Vol. 15, No. 1, pp. 8-36, 1968.

[Salton 1968] G. Salton. *Automatic Information Organization and Retrieval*. New York: McGraw-Hill, 1968.

[Salton & Buchley 1988] G. Salton and C. Buchley. Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, Vol. 24, No. 5, pp. 513-523, 1988.

[Saraclar et al. 2004] M. Saraclar, R. Sproat. Lattice-based Search for Spoken Utterance. In *Proc. HLT'04*, Boston, 2004

[Sebe et al. 2003] N. Sebe, Michael S. Lew, Arnold W.M. Smeulders. Video retrieval and summarization. *Computer Vision and Image Understanding*, 2003.

[Singhal et al. 1996] A. Singhal, C. Buckley, and M. Mitra. Pivoted document length normalization. In *SIGIR '96*: *Proceeding of the 11th annual international ACM SIGIR conference in Research and development in information retrieval*, pp. 21-29, 1996.

[Tsai et al. 2007] M.-F. Tsai, T.-Y. Liu, T. Qin, H.-H. Chen, W.-Y. Ma. FRank: a ranking method with fidelity loss. In *SIGIR '96*: *Proceeding of the 11th annual international ACM SIGIR conference in Research and development in information retrieval*, pp. 21-29, 2007.

[Vapnik 1995] V. Vapnik. *The Nature of Statistical Learning Theory*, Springer-Verlag,

London, 1995

[Wagstaff et al. 2001] K. Wagstaff, C. Cardie, S. Rogers, and S. Schroedl. Constrained K-means Clustering with Background Knowledge. In *Proceedings of the Eighteenth International Conferece on Machine Learning, ICML*, pp. 577-584, 2001.

[Xu et al. 2006] J. Xu, Y. Cao, H. Li, and Y. Huang. Cost-sensitive Learning of SVM for Ranking. In *Proc. ECML*, pp. 833-840, 2006.

[Yue et al. 2007] Y. Yue, T. Finley, F. Radlinski, and T. Joachims. A support vector method for optimizing average precision. In *SIGIR '07*: *Proceeding of the 30th annual international ACM SIGIR conference in Research and development in information retrieval*, pp. 271-278, 2007.

[Zhai & Lafferty 2001] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to Ad Hoc Information Retrieval. In *SIGIR '04*: *Proceeding of the 11th annual international ACM SIGIR conference in Research and development in information retrieval*, pp. 179-214, 2004.

[Zhou et al. 2006] Z.-Y. Zhou, P. Yu, C. Chelba, and F. Seide. Towards Spoken-Document Retrieval for the Internet lattice Indexing For Large-Scale Web-Search Architectures. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the ACL*, pp. 415-422, 2006.

[陳光華 1999] 陳光華。資訊檢索技術之核心。大學圖書館，3(1)，17-28，1999。