基於主題資訊賦予特徵不同比重之摘要系統
An Exploration of Topic-Dependent Feature-Weighting for Summary Extraction

吳家威 劉昭麟
國立政治大學 資訊科學研究所
{g9010,chaolin}@cs.nccu.edu.tw

摘要

本文主要針對英文自動文件摘要方法進行研究與探討。目前許多的摘要系統採用各種特徵選擇摘要，本文也以特徵摘要方法為主要的討論範圍。一般而言，文章的摘要與文章的主題往往有著密切的關係，文章的主題資訊常成為獲取摘要資訊的重要來源。本文將文章的主題資訊與特徵摘要的方法結合，我們認為不同主題的文章需使用不同的特徵來選取摘要，因此在使用特徵選取摘要前，先將文章依照主題分類，再給予不同主題文章不同的特徵權數，期能增進使用特徵選取摘要的正確性。本文將不同的特徵作系統性的分析，同時使用文件分類所使用的關鍵詞作為選取摘要的特徵，並提出一套利用關鍵詞獲取主題資訊進而選取摘要的方法。

關鍵字：摘要(Summary)、文件摘述(Document Extraction)、機器學習(Machine Learning)

1. 簡介

自動文字摘要系統(Automated text summarization system)是指：產生一個縮短原本文章的內容以及對於文件的基本內容進行一簡明的描述。但如何經由了解文章內容，詮釋、重新編寫摘要(Abstract)、最後產生摘要是十分困難的問題。因此，許多的研究將方向由重新編寫文章轉換成從文章的句子中圈選出摘要。例如：Luhn[7]認為自動摘要的過程即是將文章中最重要的句子選出，自動摘要的方法即是如何找出這些重要的句子。他認為通常作者會重複使用重要的字或自己習慣的用詞，用以描述主題或者這些重覆出現的字也代表著文章的主題。基於這個理由，他提出辨別最重要句子的方法在於利用最常出現的字，也就是高頻率的字作為選取摘要的工具。這個重要特徵仍被今日的許多摘要方法所採用[4]。

除了使用高頻字外，Edmundson[3]來也加入了文章的結構性特徵：段落的位置、線索字(Cue words)、標題字(Title words)作為偵測的方法。段落的位置以及線索字也廣泛的被以後的自動摘要系統所採用。透過文章的結構增加辨別摘要能力的概念也被許多以後的研究所採納並且延伸。

許多對於自動摘要的研究使用特徵來衡量文章句子的重要性。但摘要所需提供的資訊，往往隨著文章主題的不同而不同。為了能夠更精準的掌握不同性質文章所提供的不同資訊，我們先將文章依照不同的主題予以分類，再透過特徵摘要系統選出摘要。我們將利用各種的特徵值對文章擷取摘要，分析並評估每個特徵對於摘要選取的影響。選取摘要的實驗主要分為兩組，一組為經過文章分類後，再利用特徵進行摘要的選取；另一組為不經過文章分類。最後，比較並分析兩組實驗的結果。

文章的主題往往是擷取摘要的重要依據。例如：Hovy 與 Lin[4]提出的自動摘要系統 SUMMARIST 中，主題的分析即佔核心的角色。然而，許多摘要系統中主題偵測通常由高頻字所代表，例如：Kupiec 等人[6]提出利用數種特徵決定摘要的選取，Term-frequency 在 Kupiec 等人的摘要方法中用來判斷文章中所含有的主題字，段落的長度則被用來刪除過短的句子，也就是含有字數太少的句子。最後，以貝氏分類(Bayesian Classifier)的方法找出最可能的數個句子作為摘要。

對於主題資訊的獲取我們提出改進的方法，我們透過演算法找出代表各個類別的關鍵字作為主題資訊獲取的基準。關鍵字除了具有實驗所需的分辨文章主題的功能外，事實上也具有代表該文章主題的特性。在本論文的實驗中，我們將以往以高頻字作為主題字的方法改由分類所用的關鍵字取代，並分析這兩種方法的實驗結果。

本論文將採用以下介紹順序：第二節：資料來源、第三節：特徵摘要方法描述、第四節：利用各種方法分析摘要特徵、第五節：利用統計方法與決策樹將特徵整合、第六節：結論。

2. 資料來源

實驗的資料分別來自 Wall Street Journal、USA Today、New York Times 的新聞資料：一共收集 416 篇新聞，根據文章的主題區分為 13 種類別。新聞的分類依據 ProQuest 線上資料庫查詢所得到。由於 ProQuest 中所取得的新聞資料皆附有摘要，其選出的摘要段落便是我們

用作訓練資料以及評估系統的依據。

3.特徵摘要方法描述

本節將描述本摘要系統所採用的特徵，以及取得特徵的方式：

3.1 主題關鍵詞：

過去的研究中，許多的摘要系統使用高頻率的詞作為文章的主題詞。此做法仍然有其缺點。例如：若兩個詞有相同的意義或者能代表相同的主題，但卻因為在選取高頻詞時，遺漏了其中一個次數較少的詞，造成主題資訊獲取的偏差。另外一個問題是究竟我們應取幾個高頻詞作為主題的關鍵詞，選取的個數往往會影響到最後的結果。上述的問題其原因在於：我們並不知道究竟哪些詞代表主題，我們只能假設文章中選出的高頻詞皆為重要的主題詞，為了改善此一情況，我們事先將每個類別的主題詞利用文件分類演算法挑出，以下是我們的做法：

3.1.1 將文章切分成單一主題的段落

一般文件分類的做法是將文章直接作為分類的基本單位，若訓練資料中的文件含有多個主題(Multi-topics)則忽略不列入考慮，因為將一篇超過一個主題的文章直接置入文件分類的演算法中，將會造成分類結果的偏差。我們的資料來源皆為新聞文章，一篇新聞的內容往往不只包含一個主題，而是許多議題的綜合。若我們事先忽略含有多種議題的文件，對於自動摘要的實用性會有非常嚴重的影響，將導致大部分的新聞資料皆無法使用。因此，我們的做法以段落作為基本的單位，每一段落會被視為一篇文章，供文件分類的演算法使用。

I. 將每個類別的每篇文章的三個大寫字選出，作為初步文件分類的關鍵詞：

首先，統計每篇文章的大寫字出現次數，選擇每篇文章中出現最多次的三個大寫字，目的是希望能找出每一篇文章中的關鍵詞。同時我們也假設這些關鍵詞與文章的主題是一致的，這些選出來的詞將作為各個主題的關鍵詞，做初步分類工作。

II. 將文章的段落分出，並為每一個段落標上所屬類別

我們利用前一步驟每個類別所擁有的關鍵詞，辨別每個段落所含的主題。我們統計每一個段落的詞與每個類別所擁有關鍵詞的共同出現次數；接下來選出與段落共同出現次數最多的類別，作為該段落的主題。最後我們將每個段落當作一篇文章並擁有一個主題。

3.1.2 Clustering words 演算法的數學背景介紹

我們採用 Distributional Clustering 作為分類的方法，所使用的演算法主要利用詞在不同主題文章出現的分佈情況作為分類的依據，以下的 Distributional Clustering 演算法採自[1]，對演算法的說明如下：

首先，c_j 為第 jth 種文章類別，C={ $c_1, c_2 ... c_m$ }，我們有 m 種主題，D={ $d_1, d_2 ... d_n$ }代表文件的集合。此時，我們可以將文件視為一連串文字的組合且彼此之間是獨立的。因此，$P(d_i \mid c_j)$的條件機率可視為是一串連續的詞出現機率相乘。$P(d_i \mid c_j)$可以寫成下式：

$$P(d_i \mid c_j) = \prod_{k=1}^{|d_i|} P(w_{ik} \mid c_j)$$

w_{ik} 表示第 ith 篇文章的第 kth 個詞，$|d_i|$ 表示文章 d_i 的詞數。此時我們需要用收集的樣本去估計需要的參數。我們假設此模型的參數為 $\hat{\theta}$。我們用 $\hat{\theta}_{c_j}$ 表示第 j th個類別的事前機率估計值，也就是 $\hat{\theta}_{c_j} \equiv P(c_j)$。假設每一篇文章的出現機率為 $P(d_i)$，$P(d_i) = \frac{1}{|D|}$。其中 $\hat{\theta}_{c_j}$ 的計算方式如下：

$$\hat{\theta}_{c_j} \equiv \left(\frac{\sum_{i=1}^{|D|} P(c_j \mid d_i)}{|D|} \right)$$

$P(c_j \mid d_i)$ 將利用現有的資料取得，方法為：當 $d_i \in c_j$，當 d_i 屬於 c_j 類別時，$P(c_j \mid d_i)$ 為一，若文章不屬於該類別時則為零，例如：d_1 屬於 c_1 類別則 $P(c_1 \mid d_1)$ 為 1；d_1 不屬於 c_1 類別 $P(c_1 \mid d_1)$ 為零。將全部屬於該類別的文章次數加總除以總文章數即可得到該類別的文章出現的機率。

現在我們必須估計給定一類別後，每個詞出現於該類別的機率，我們使用 w_t 代表詞的集合，$P(c_j \mid d_i)$ 則用 $\hat{\theta}_{t|c_j}$ 來估計，$P(w_t \mid c_j)$表示在已知為 c_j 類別的情況下，w_t 出現的機率。$N(w_t, d_i)$表示 w_t 在 d_i 文件出現的次數，$P(c_j \mid d_i)$ 與前述相同 $d_i \in c_j$ 則為一，反之則為零。將 w_t 與屬於 c_j 類別的文章所共同出現的次數加總，除以所有詞與類

別共同出現的加總來估計 $\hat{\theta}_{t|c_j}$，其中 w_s 表示在全部文章中曾出現的詞，如下式：

$$\hat{\theta}_{w_t|c_j} \equiv \frac{0.1+\sum_{i=1}^{|D|} N(w_t,d_i)P(c_j \mid d_i)}{0.1*|V|+\sum_{s=1}^{|V|}\sum_{i=1}^{|D|} N(w_s,d_i)P(c_j \mid d_i)}$$

$|V|$ 表示全部文字的長度，分子的 0.1 是為了作 smoothing[9]。我們經過實驗發現，將分子的加一改成加 0.1 效果較好，因此改成 0.1，本節的完整式子可參考[1]。

3.1.3 Clustering Words 演算法的步驟

在上述的數學模型背景下，假設我們共有 n 篇文章，有 V 個不同的詞，m 個類別。演算法的主要目的在於將相近的詞置於同一類別中，相近的定義為詞出現於各種主題文章的次數分佈的接近程度。我們將詞依序置入 slot(用來存放詞彙的空間)中，再將近似的 slot 合併達到分類的效果。文件分類的演算法如下：

1. 將 V 個詞依據主題的分佈狀況排列，分佈越集中，排序越前面，我們利用 Entropy 衡量詞彙在類別中的分佈狀況。

由於文章的分類是利用每個詞在文章之間的分佈，若此時一個分佈很廣的詞，也就是出現於各種主題的詞，於演算法一開始時就進入 slot 中，會造成此詞彙與大部份的詞相似度皆類似，因此，我們需要將文字的分佈較為集中的詞挑出。我們利用之前得到的 $\hat{\theta}_{w_t|c_j}$ 根據貝式定理轉換成 $\hat{\theta}_{c_j|w_t}$，其中所需的 $P(c_j)$ 與 $P(w_t)$ 已於之前求得，計算其資訊含量(Entropy)[9]，公式如下：

$$Entropy\ (w_t) = -\sum_{j=1}^{m} p(c_j \mid w_t)\log\ p(c_j \mid w_t)$$

2. 將V中排序在前面的m+1個詞置入m+1個 slot 中。

此步驟為初始化階段，每一個 slot 皆需要一個以上的詞才能計算出 slot 本身的分佈。

3. 計算每一個 slot 的分佈情況：

經過 Entropy 排序過的詞依序放入 slot 後，每一個 slot 都至少存了一個詞。我們要將兩個最相近的類別合併，就必須知道每個 slot 的分佈。所謂的分佈定義為：已知一個詞為 w_t，在第 j 個類別出現的機率，也就是 $P(c_j \mid w_t)$。我們將每個 slot 中詞的分佈加以平均，代表 slot 的分佈。平均的公式如下：

$$P(c \mid w_{slot}) = \frac{\sum_{t=1}^{|V_{slot}|} P(w_{slot,t})P(c \mid w_{slot,t})}{\sum_{t=1}^{|V_{slot}|} P(w_{slot,t})}$$

w_{slot} 表示存入某一 slot 中的詞組，共有 $|V_{slot}|$ 個詞。我們將此 slot 中的每一個詞出現於 c 類別文章的機率平均，利用每個詞的相對出現次數作平均。此步驟必須對每一個 slot 重複作 m 次，因為每一個詞皆有出現於所有類別的機率，同樣的每一個 slot 也必有對每一個類別的出現機率，每一個 slot 都會有 $P(c_j \mid w_{slot})$，j=1….m。

4. 選擇兩個相近的類別予以合併：.

取得每個 slot 的分佈狀況後，我們將比較任兩個 slot 的相似度，相似度的原則是：若兩個 slot 於相同類別的出現次數越相近，表示這兩個 slot 的詞越可能是同一類。我們使用 KL-divergence[1]衡量兩個類別的近似度，如下面的公式所示：

$$D(P(c \mid w_{slotA}) \| P(c \mid w_{slotB})) = \sum_{j=1}^{|c|} P(c_j \mid w_{slotA})\log\left(\frac{P(c_j \mid w_{slotA})}{P(c_j \mid w_{slotB})}\right)$$

w_{slotA} 表示 slotA，w_{slotB} 表示 slotB。事實上，我們是希望將兩個 slot 在相同類別的出現次數最相近者找出，因此我們必須比較兩個 slot 個詞對應類別的機率，我們採用 KL-divergence 衡量兩個 slot 的近似度。

5. 從 V 中取下一個詞置入空類別中，

上一個步驟完成後會有一個 slot 的是空的，我們便將下一個詞填入空的 slot 中。

6. 處理完 2/3 的詞後，演算法停止

演算法的停止條件設在全部詞數的前 3/5，原因在於只有前 3/5 的詞的分佈較為集中，適合將其置入一個類別。剩下的詞分佈過廣不適合置入一個特定的類別，這種類型的詞被 Yarowsky[11] 稱為 topic-independent distinctions。同時我們也設了一個停止條件為 V 的詞的 Entropy 值大於 0.5 時演算法停止。

在結束 clustering words 的演算法後，我們得到每個 slot 所存放的詞代表某一個類別。但我們仍然不知道哪一個 slot 對應到哪一個類別。所以我們利用每一個 slot 的 $c_j \mid w_{slot}$ 值，選出一個 slot 的 $c_j \mid w_{slot}$ 中，j=1….m，哪一個 c_j 使 $c_j \mid w_{slot}$ 的值最大，得到的 c^* 便是該 slot 所代表的類別，如下式。

$$c^* = \arg\max_{c_j} p(c_j \mid w_{slot})$$

此時，每一個 slot 將會對應一個主題，而 slot 中所存的詞即為代表該主題的關鍵詞，這些關鍵詞以下稱為 clustering words。

3.1.4 利用 Clustering Words 選取摘要

接下來介紹我們如何使用 clustering words 作為選取摘要的特徵。我們希望利用 clustering words 將文章中出現的詞結合成一個較高層次的概念(Concepts)，也就是文章的主題，將純粹只計算字數的方式轉變成對於概念的計算。在此章中，每一個類別代表一個主題，因此類別與主題的意義在此事共通的。以下是我們的方法：

我們的做法是先統計一篇文章的詞與每個主題所包含的 clustering words 的共同出現次數：

$$s_j = N(t_j, d) \qquad \forall j,\ j = 1 \cdots |T|$$

s_j 表示該文章中第 j[th] 個主題所得到的分數，$N(t_j, d)$ 為第 j[th] 主題與該篇文章所共同出現次數。$T = \{t_1, t_2, ...t_m\}$ 表示主題的集合，t_j 第 j[th] 個主題的關鍵詞集合，$|T|$ 代表總共的類別數。通常文章可能不只擁有一個主題，我們必須決定文章所包含的主題為哪些。我們的方法是使用 k-means 演算法[2]將所有的主題的比重切割成顯著的主題與不顯著的主題，以下是我們的說明：

我們將欲區分的類別分為兩種：顯著的主題與不顯著的主題。主題若為顯著的，其 clustering words(該主題的關鍵詞)與文章的詞的共同出現次數(上式的 s_j)，出現次數應該較多，而不顯著的主題應該較少。我們將其視為兩種類別，使用 k-means 區分此兩類別。我們將 k-means 所需的相似度公式定義為 $|s_j - m|^2$，也就是共同出現的次數與該類別的共同出現次數平均數的距離，m 表示 mean。最後，將被分為顯著的主題群，視為文章所包含的主題，並利用此結果進行對摘要的分析。

在取得文章的主題之後，接著便分析每一個段落與文章主題的相似度。衡量相似度的方法是：計算每個段落與文章顯著主題所擁有關鍵詞的共同出現次數的加總，相似度同時是選取摘要的計分方式。但若只計算單純的字數出現次數可能造成較重要的主題被次重要的主題所取代，因此不能只計算每個類別的詞於一篇文章的出現次數，在計算分數的同時必須反映其主題之間的比重，也就是區隔主題之間不同的重要性，因此必須乘上主題的比重作為每個詞出現的權數。權數的計算方法如下面公式所述：

$$s_j = \sum_{k=1}^{|P|} N(t_j, p_k) \qquad \forall j, j = 1 \cdots |T|$$

s_j 表示 t_j 主題在文章中所佔的比重，$N(t_j, p_k)$ 為 t_j 的關鍵詞與 p_k 段落的詞的共同出現次數。$|P|$ 為該文章所擁有的段落數。

$$Score(p_k) = \sum_{j=1}^{|c|} \left[S(t_j, p_k) * \sum_{k=1}^{|p|} N(t_j, p_k) \right]$$

$Score(p_k)$ 為每一個段落的分數，$S(t_j, p_k)$ 由前面的式子所得到，也就是 s_j，表示每個類別在文章中所佔的比重與類別的權數，以反映主題之間不同的重要性。最後，計算每一個段落的分數，依照分數將段落排列，依序選出一定比例的段落作為選出的摘要。

應用此方法的缺點在於：文章的主題可能判斷錯誤，若此情形發生時將會導致將新聞文章套用於錯誤的主題上，以致摘要擷取的方向錯誤。為了避免這種情況發生，我們會比較高頻率的詞與 clustering words，若出現次數最多的高頻詞與我們選出代表主題的關鍵詞完全不相同，我們就認為文章主題的辨認發生錯誤。我們相信高頻詞能夠代表主題詞，但並非全部的高頻詞。例如：選出文章中出現最多次的五個詞，這五個詞至少一個詞包含重要的主題詞的機會很大，這並不意味這其他四個詞皆有足夠主題代表性，這也是我們用 clustering words 的重要原因。

其他的特徵包含有：高頻詞：高頻詞即為出現次數較多的詞。日期：統計每個段落中日期相關詞出現的次數作為分數。數字：將出現的次數作為段落的分數加以排序。專有名詞：專有名詞通常對於文章的主題具有著重要的意義。為了偵測專有名詞我們採用大寫字作為代替。段落的長短：段落的長短表示段落所含的詞數的多寡。實驗的方法是將段落依據其長度作為排序的依據，依序選出一定的比例作為摘要，並檢視其結果。

4. 特徵分析

尋找用於選擇摘要的有效特徵是許多研究努力的目標，目前為止有許多的特徵被用於選取摘要也獲得了不錯的成果。我們將針對所選出的特徵做系統性的分析，分析各項特徵的目標

在於了解各別特徵對於摘要的選取能力，以及不同主題的文章對於特徵的影響。

4.1 使用個別特徵挑選摘要的實驗與結果

本節將利用 3.1~3.4 節所介紹的特徵與選取摘要的方法進行摘要的選取。而評估方法如下：依照各個特徵值選出一至九個段落作為摘要，並使用下列的公式計算其精準度(Precision)與召回率(Recall)。[9]

$$\text{Precision}=\frac{J}{K},\ \text{Recall}=\frac{J}{M},$$

$$\text{F-measure}=\frac{2}{\frac{1}{precision}+\frac{1}{recall}}$$

J 為選取的段落且同時為正確的摘要的數目，K 為一共選取幾段作為摘要，M 為正確的摘要共有幾個段落。實驗的結果列於圖一：

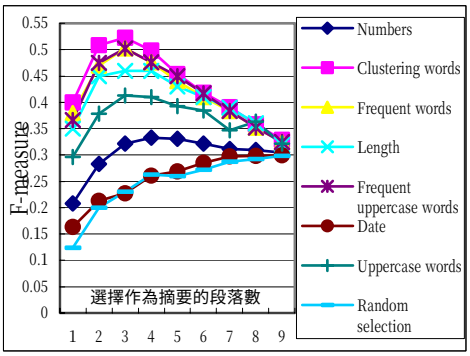


圖 1 利用各種特徵值選取摘要的 F-measure

圖一中的 random selection 表示隨機選取段落作為摘要，目的在於評估摘要系統的最低表現(Baseline Performance)。我們可以發現 clustering words 的特徵值、frequent words 與其中 clustering words 利用主題的關鍵詞獲取主題資訊，對於文章摘要的選取的表現較好。

4.2 利用 Information Gain 分析特徵的摘要選取能力

接下來我們希望能獲取每個特徵對於不同主題文章的摘要選取能力，我們將利用 information gain 作為分析的指標。我們利用每一個特徵將區分好類別的文章加以分析，將所有段落分為摘要段落與非摘要段落分成兩類。我們舉一個例子，若原本有摘要段落有十段，非摘要段落 50 段，利用 frequent words 區分摘要之後，被分為摘要與非摘要，但被分成摘要的十個段落中有五個是正確的摘要五個是錯誤的，而非摘要的部分有五個是真正的摘要段落，其餘 45 個段落是正確的分類。而後以上述的數據計算特徵所獲得的 Information，將未分類之前所獲得的資訊量扣除分類後的資訊量即為最後該特徵所獲得的 information gain[9]。請見下面公式說明：

$$Gain(A_k,S_i)=H(O_i)-\sum_{j=1}^{2}\frac{|S_{ij}|}{|S_i|}H(S_{ij})$$

H 表示 Entropy，S_i 表示第 i[th] 類主題的文章，A_k 表示第 k[th] 個特徵，O_i 表示未經該特徵分類原本的分類情況。j 表示摘要與非摘要的段落。最後我們將結果顯示於圖 2，由圖 2 所示可以發現各項特徵對於不同主題的區分能力有很大的差異，這表示對於不同主題的文章，特徵對摘要的區分能力有很大的差異、甚至有些特徵對於某些主題的摘要選取是完全不具有鑑別能力。

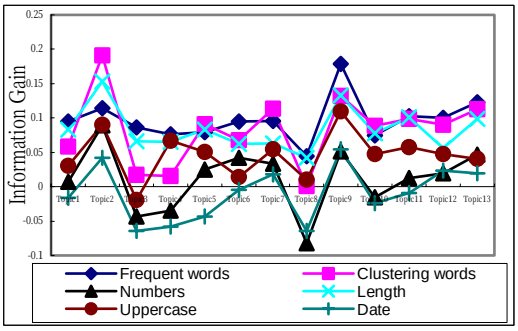


圖 2 各種特徵於不同主題的文章所獲得的 Information Gain

例如：數字特徵值，我們發現數字對於 Topic5 、Topic6、Topic7 與 Topic9 的摘要選取能力較高，但對於 Topic 3、 Topic 4、 Topic 8、Topic 10 是不具有選取摘要能力。因此，對於 Topic 3、 Topic 4、 Topic 8、Topic 10 我們不應該使用數字作為選取摘要的手段。但 information gain 在某些主題中呈現共同的走向，所有的特徵皆很高或很低，若將此因素考慮進去的話，數字特徵真正表現最好的為 Topic 5 與 Topic 6。date 這項特徵在六個 Topics 中呈現負的 information gain 但對於四種 Topics 的摘要選擇能力是正的。其餘的特徵皆具有一定程度的摘要選取能力，但對於不同主題文章的選取能力仍有一定的差距。

4.3 利用段落特徵值的排序分析

此節我們將每個摘要段落依照不同特徵值含量所取得的分數排序，分成十等份。例如：一篇文章有十個段落，對於高頻詞這個特徵而言，這十個段落分別有不同的高頻詞出現次數，此時這十個等份的段落便依照此順序排列。排名於前百分之十的段落表示含有的高頻詞次數最多；排名最後則表示含有該特徵的值最少。此時記錄摘要段落排序的百分比，待全部的文章皆完成統計後，統計排序十等份中的每一等份，有幾個摘要段落置於其中，將每一

等份的摘要段落數除上全部的摘要段落數，得知摘要段落於每一等份的出現比例。結果列於圖 3。

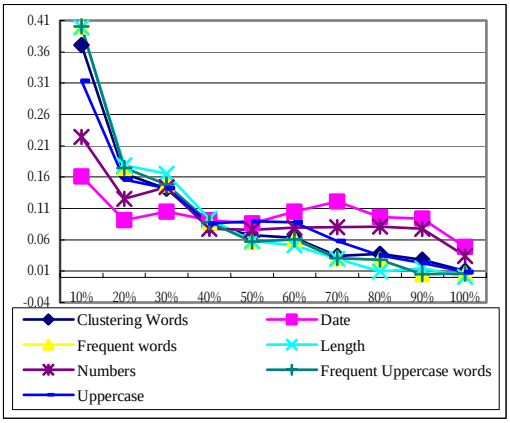


圖 3 摘要段落的特徵值排序

由圖 3 可以發現：除了 date、numbers 以外的特徵，大部分的摘要段落所獲得的分數排序較高。此外，有接近四成的摘要段落的 clustering words、frequent words 與 frequent uppercase words 三個特徵的排序為前 10%。這個結果對於摘要的選取是有幫助的，我們認為摘要的段落的選擇至少必須有一個段落以上，是直接用來描述該文章的主題，而三個特徵的目標皆是欲獲取主題的關鍵詞，並皆有一定的主題獲取能力，因此當一個段落含有最多的主題詞時，被選為摘要的機率也越高。

5 實驗

接下來我們使用兩種方法將各種特徵方法整合找出文章的摘要，第一種方法是統計方法，第二種方法是決策樹：

5.1 統計方法

4.3 節利用特徵值將摘要段落依次排序以獲得不同特徵出現機率，此機率可視為當已知段落為摘要段落時的特徵表現，我們利用此機率作為選擇摘要的基礎。以下將說明我們的做法：

$P(s)$ 為某一段落被列為摘要的機率，f_n 為第 n 個特徵值，$P(f_1, f_2, \cdots f_n \mid s)$可解讀為已知該段落為摘要時 $f_1, f_2, \cdots f_n$ 發生的機率。

$$P(s \mid f_1, f_2, \cdots f_n) = \frac{P(f_1, f_2, \cdots f_n \mid s)P(s)}{P(f_1, f_2, \cdots f_n)}$$

此時假設在已知段落是否為摘要的情況下，每個特徵的出現為獨立的，我們可以把上式改寫為：

$$P(s \mid f_1, f_2, \cdots f_n) = \frac{P(s)\prod_{j=1}^{n} P(f_j \mid s)}{\prod_{j=1}^{n} P(f_j)}$$

在比較段落的過程中，$P(s)$為該篇文章出現摘要段落的比例，同一篇文章中的每個段落 $P(s)$為相同的值，因此我們可以將其 $P(s)$消去。$\prod_{j=1}^{n} P(f_j)$表示一段落的特徵表現排序發生的機率，在不給定該段落是否為摘要時，每個段落出現特徵值的機率是相同的，原因在於我們將排序分成等距的十等份，例如：一段落的特徵排序為第一，但不給定其是否為摘要段落，此時無論其特徵的表現排名為何，其機率與其他等份的出現機率相同。因此，在比較同一篇文章段落的過程中，每個段落所獲得的 $\prod_{j=1}^{n} P(f_j)$的值是相同的。因此，$\prod_{j=1}^{n} P(f_j)$在比較的過程中也能夠消去。我們將每個段落的計算結果作為其得到的分數。上式可進一步簡化為：

$$Score(s) = P(s \mid f_1, f_2, \cdots f_n) = \prod_{j=1}^{n} P(f_j \mid s)$$

$P(f_j \mid s)$為我們統計摘要段落的特徵排序值，計算的過程已經於 4.3 節說明。經過文章分類的實驗必須先經判斷決定文章的主題，我們採用的文章分類公式如下：

$$c^* = \arg\max_{c_j} \cos(\vec{c}_j, \vec{d}) = \frac{\vec{c}_j \cdot \vec{d}}{|\vec{c}_j||\vec{d}|}$$

$\vec{c}_j$ 表示 c_j 的類別中的主題關鍵詞，$\vec{d}$ 表示 d 文章中所出現的詞，我們希望找出一個 c^* 使 c_j 的關鍵詞與 $\vec{d}$ 中的文字的 $\cos(\vec{c}_j, \vec{d})$值最高，使其最高的類別便是文章的主題。

事實上，並非所有的特徵對於摘要的辨別能力皆相同，因此我們需要將不同特徵的貢獻度予以反映，以避免選取摘要時因為使用不具有辨別摘要的特徵造成選取的偏差。我們利用之前計算出每個特徵的 information gain 作為每個特徵的權數。Information gain 的獲取需視實驗而決定，當實驗為文章先經過分類時，information gain 也會經由分類過的文章所取得，若實驗為不分類時，information gain 就會由不經分類的方式獲取。

使用權數表示先將段落的計分公式取對

數，將相乘轉換為相加，如下面公式所述：

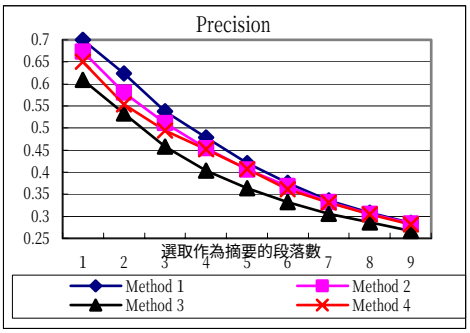
$$Score\ (s) = \sum_{j=1}^{n} \log\left(P\left(f_j \mid s\right)\right) * w_i$$

w_i 為第 i 個特徵的 information gain。若文章經過分類時，w_i 則改寫成 w_{ik}，k 為該篇文章分到哪一個類別而決定。在簡單的計算過程後可以得到每個段落的分數，依分數的高低排序，依次選出段落作為摘要。我們將採用四種不同計分的方法來選取摘要：

第一種方法(Method 1)為使用公式(1)的計分方式，即加入 information gain 的方法，且文章經過主題的分類。第二種方法(Method 2)為使用公式(1)的計分方式，但文章不經過分類。

$$Score\ (s) = \sum_{j=1}^{n} \log\left(P\left(f_j \mid s\right)\right) * w_i \quad (1)$$

第三種方法(Method 3)為文章經過分類，但計分方式不使用 information gain，也就是公式(1)不採用權數(w_i)的情形下，將 $\log\left(P\left(f_j \mid s\right)\right)$ 最後的加總即段落所得到的分數。第四種方法(Method 4)為不分類且不使用 information gain，即採用第二種方法的方式計分，但文章不經過分類。本實驗直接將 4.3 節所作的訓練資料直接作為實驗的測試資料：四種方法的實驗結果如圖 4 與圖 5 所示：



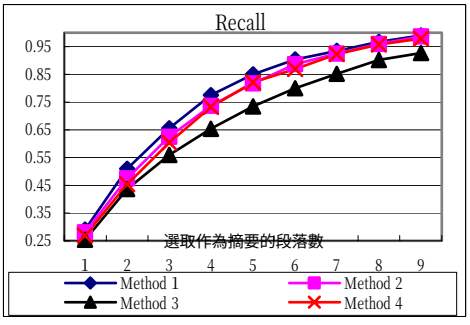


圖 4 四種不同方法所得到的 Precision 值與 Recall 值

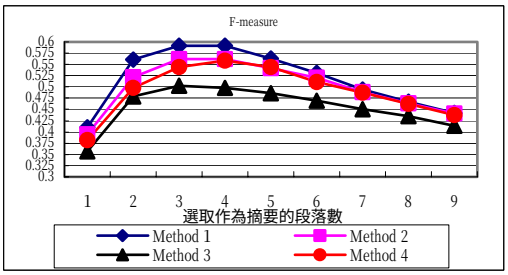


圖 5 四種不同方法 F-measure 值

由實驗結果可以發現 Method 1 無論在 Recall 或 Precision 的表現皆較其他的方法表現優異，這說明將文章依照不同性質分類對於擷取摘要的正確率有正面的助益。

若比較 Method 1 與 Method 2， Method 2 並未將不同性質文章的特徵表現區分，所統計出的機率是將全部主題的文章的特徵值表現平均。例如：一個主題的文章有 60%的摘要段落的 A 特徵表現在全部段落的前 10%，另一主題有 35%的摘要段落 A 特徵表現在全部段落的前 10%，由於每個主題的文章數一樣，因此將兩個主題的文章合併後變成 47.5%的摘要段落 A 特徵的表現在前 10%，這對兩個主題的文章皆是錯誤的這也是將文章分類後重要的好處，可能造成的選取偏差。Method 3 雖然將文章分類但並未區分不同特徵的重要性，此舉使文章的分類受到較無鑑別力的特徵所影響，對其正確率造成影響。

此外，我們可以發現 F-measure 的值在選取第四段後便開始下滑，這與我們的實驗資料有關，我們的實驗資料摘要段落數最多不超過 3 段，因此在第四段之後便逐次的下滑。

5.2 決策樹

我們採用決策樹作為選取摘要的實驗方法。實驗的系統平台採用 Weka[12]所提供的軟體，演算法為 C4.5[10]。若實驗為文章不分類時，總共有 6024 個段落，當實驗為文章分類時每個類別平均有 464 個段落進行實驗。最後便利用這些資料計算出決策樹，圖 6 為 Topic 1 文章的訓練資料所得到的決策樹。

實驗與之前類似，將分為以下兩組：方法一為文章依照主題類產生各自的決策樹，方法二為文章不經過分類，所有的文章共同產生一決策樹。評估的方式採用 10-fold cross-validation。分別計算 Precision、Recall 與 F-measure，最後的實驗結果如表格 1 所示：

表格 1 決策樹的實驗結果

	Precision	Recall	F-measure
Method 1	**0.556**	**0.417**	**0.477**
Method 2	**0.593**	**0.306**	**0.404**

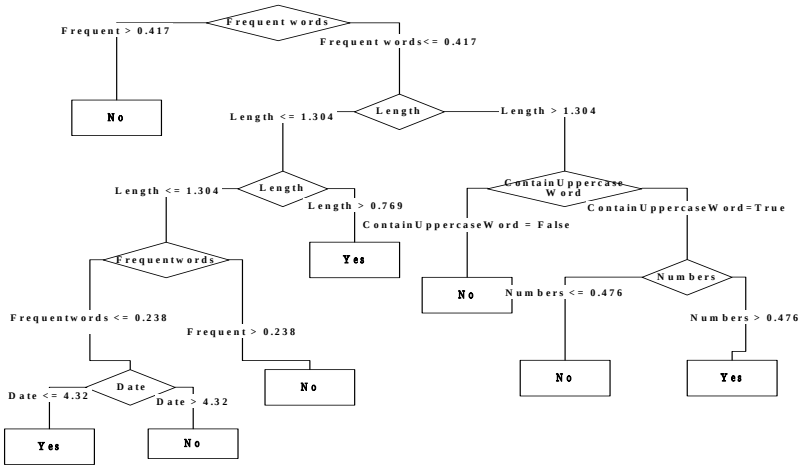


圖 6Topic 1 的決策樹

整體而言，無論 Method1 與 Method2 的表現皆不理想，其中 Method 2 的 F-measure 只有 0.4 左右，比起單獨使用一特徵值選取摘要的表現要差。與 5.1 的統計方式實驗結果相同的是經過文章分類的表現仍較未經文章分類的部分要高出許多，雖然 Precision 的值 Method 2 較高，但只高出了 3.7 個百分點，而 Method 1 的 Recall 值卻高出了 11.1 個百分點。

對於這個實驗結果我們認為：由於決策樹使用階層性的決定方式，在高頻詞或 Clustering words 等區分能力較一般性的特徵區分過後，剩下特徵的區分力較低，容易造成分類的錯誤。相對於決策樹的摘要選取方式，統計方法的優點在於能夠將擁有越多明顯特徵的段落分數越高，而非指依靠單一的特徵進行摘要的區分，因此表現也較好。

6 結論

本文主要描述利用不同的特徵找尋摘要的過程，並對每一個特徵進行系統性的分析。此外，我們也使用 clustering words 作為找尋摘要的特徵之一，提出利用關鍵詞獲取主題資訊的摘要方法，並達到了一定的效果，在加入主題檢查的情況下，選取摘要的正確率超過了我們所選用的其他特徵。在特徵的整合方法上，我們利用統計方法與決策樹方法將選用的特徵結合，並證明文章經由主題分類所達到的效果，確實較不經主題分類的摘要方法擁有較好的表現。未來我們希望能以特徵組合的方式，進一步探討特徵對於摘要的選取能力。此外，我們希望能加入文章結構的分析幫助我們對文章摘要的掌握能力。

致謝

感謝國科會研究計劃 NSC-91-2213-E-004-013 對本研究的部分資助。

參考文獻

[1] L. D. Baker and A. K. McCallum, " Distributional Clustering of Words for Text Classification," *Proc. of the 21th ACM SIGIR Conference on Research and Development in Information Retrieval*, pp.96-103, 1998.

[2] R. O. Duda, P. E. Hart, D. G. Stork "Pattern Classification and Scene Analysis: Pattern Classification," Brooks/Cole Wiley, John & Sons, Incorporated Pub Co, 2000.

[3] H. Edmundson. "New Methods in Automatic Abstracting," *Journal of ACM*, 16(2), pp.264-285, 1969.

[4] E. Hovy, and C. Y. Lin, "Automated Text Summarization in SUMMARIST, " *Advances in Automatic Text Summarization*, MIT Press, pp.81-94, 1999.

[5] H. Jing, R. Barzilay, K. McKeown, and M. Elhadad., "Summarization Evaluation Methods: Experiments and Analysis," *In Working Notes of the AAAI-98 Spring Symposium on Intelligent Text Summarization*, pp.60-68, 1998.

[6] J. Kupiec, J. Pedersen, F. Chen, "A Trainable Document Summarizer," *Proc. of the 18th ACM SIGIR*, pp.68-73, 1995.

[7] H. P. Luhn., "The automatic creation of literature abstracts," *IBM Journal of Research and Development*, 2(2), pp.159-165, 1958.

[8] T. M. Mitchell, "*Machine Learning*, " The McGraw-Hill Companies, 1997.

[9] S. H. Manning, "*Foundations of Statistical Natural Language Processing*", MIT Press, 1999.

[10] J. R. Quinlan. , "*C4.5: Programs for Machine Learning*, " 1993.

[11] D. Yarowsky, "Word-Sense Disambiguation Using Statistical Models of Roget's Categories Trained on Large Corpora," *Proc. of the 14th International Conf on Computational Linguistics*, pp.454--460, 1992.

[12] http://www.cs.waikato.ac.nz/ml/weka/