第 3 章 自動摘要模型

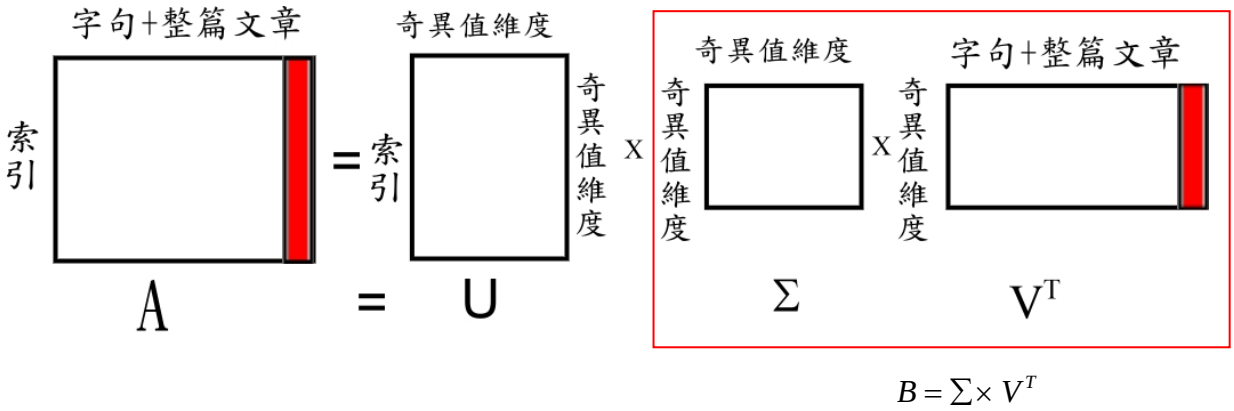
如何才能從文件中自動擷取出重要的字句，以之做為整篇文件的摘要，這是自動摘要所要探討的問題，本論文提出嵌入式潛藏語意分析模型、隱藏式馬可夫模型、主題混合模型等自動摘要模型，茲將其分別說明如下各小節。

3.1 嵌入式潛藏語意分析（Embedded LSA）模型

基於對潛藏語意分析與向量空間模型的探討，本論文提出嵌入式潛藏語意分析模型，其將每一字句與整篇文件共同投影到潛藏語意空間，最後藉由向量空間模型，估測各字句與整篇文件的相關性，演算法如下：

1. 將文件 D 斷句，$D = \{S_1, S_2, ..S_i.., S_N\}$
2. 由文件 D 建立 索引×字句矩陣 A，並將整篇文件嵌入到矩陣的最後一行
3. 對 A 進行奇異值分解，得到左奇異向量矩陣 U、奇異值矩陣 $\sum$ 與右奇異向量矩陣 V^t
4. 在右奇異向量矩陣 V^t 中，最後一行向量即為整篇文件在語意空間的表示法，其餘行向量即為原始文件中各字句在語意空間的表示法，將 $\sum$ 與 V^t 相乘得到各字句與整篇文件在潛藏語意空間的投影（$B = \sum \times V^T$）
5. 將 B 的最後一行（即整篇文件的投影）與 B 中的其他行向量（各字句），以向量空間模型表示，並進行餘弦相關度估測，得到一句排名
6. 依摘要比例，將句排名所對應的字句，摘錄形成摘要

如圖 3.1 所示，紅色部份即為所嵌入的整篇文件，*矩陣 B* 最後一行向量即為整篇文件的投影，將其與其他行向量（字句）做餘弦相關度估測後，得到一句排名，用以依摘要比例摘錄形成摘要。



$$B = \sum \times V^T$$

圖 3.1 嵌入式潛藏語意分析模型示意圖

3.2 隱藏式馬可夫模型-型一（HMM-Type1）

近年來有學者提出 HMM/N-gram based Model 用於中文語音文件檢索上 [Chen *et al.* 2004a]。延伸其應用，視文件為一機率生成模型（Probabilistic Generative Model），對於每個索引都有一對應的機率分佈，文件與文件中每一字句的相關性，是藉由每一字句的所有索引在文件發生的相似值（Likelihood）來決定，也就是說當字句的索引在文件的機率分佈值連乘積越高，則字句與文件的相關性就越高，如圖 3.2 所示，數學式如下：

$$p(S_i|D) = \prod_{w \in S_i} p(w|D) \approx \prod_{w \in S_i} \left[\lambda p(w|D) + (1-\lambda) p(w|Corpus) \right] \quad (3.1)$$

其中 $p(w|D)$ 為文件 D 產生索引 w 的機率值，並與一更大語料庫做平滑化（Smooth），$p(w|Corpus)$。

演算法如下：

1. 將文件 D 斷句，$D = \{S_1, S_2, ..S_i.., S_N\}$
2. 計算文件 D 的單連語言模型
3. 對文件 D 中各字句 S_i 估測 $p(S_i|D) \approx \prod_{w \in S_i} \left[\lambda p(w|D) + (1-\lambda) p(w|Corpus) \right]$

   機率值，並依此做排序形成一句排名
4. 依摘要比例，將句排名所對應的字句，輸出形成摘要

假設在一篇文件中的索引，其重要性皆相同，愈長的字句其分數愈低，是以

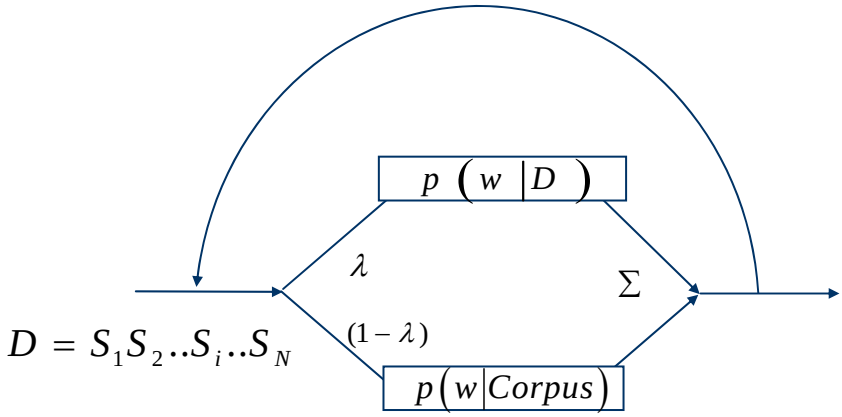


圖 3.2 隱藏式馬可夫模型-型一示意圖

在估測文件產生每一字句的機率 $p(S_i \mid D)$ 時，以每一字句長度分之 1 為次方對分數開根號（正規化），$\sqrt[|S_i|]{p\left(S_i\middle|D\right)}$，以避免句長影響到選取摘要字句的正確性。

此外，對於每一個文件，將文件 D 視為與自己相關，則參數 λ 與文件 D 產生各索引 w 的機率值可藉由期望值最大化演算法 [Dempster *et al.* 1977]，自動調整參數與訓練模型，數學式如下所示：

$$\hat{\lambda} = \frac{\sum_{w \in D} E\left(w, D\right)}{\left|D\right|} \tag{3.2}$$

$$\hat{p}(w \mid D) = \frac{E\left(w, D\right)}{\sum_{w \in D} E\left(w, D\right)} \tag{3.3}$$

$$E\left(w, D\right) = n\left(w, D\right) \frac{\lambda\, p\left(w\middle|D\right)}{\lambda\, p\left(w\middle|D\right) + \left(1-\lambda\right) p\left(w\middle|Corpus\right)} \tag{3.4}$$

其中 $\left|D\right|$ 是文件 D 的長度，$n\left(w, D\right)$ 是索引 w 出現在文件 D 的次數。

更進一步來說，文件 D 中每一字句 S_i 可利用與其相關的字句 $\widehat{S_i}$（可由字句 S_i 與一斷句後的文件語料庫，經由餘弦估測其相關度，最後再選取最相關的字句組成 $\widehat{S_i}$），做字句擴充（Sentence Expansion），如下所示：

$$p\left(\widehat{S_i}\middle|D\right) = \prod_{w \in \widehat{S_i}} \left[\lambda\, p\left(w\middle|D\right) + (1-\lambda)\, p\left(w\middle|Corpus\right)\right] \tag{3.5}$$

3.3 隱藏式馬可夫模型-型二（HMM-Type2）

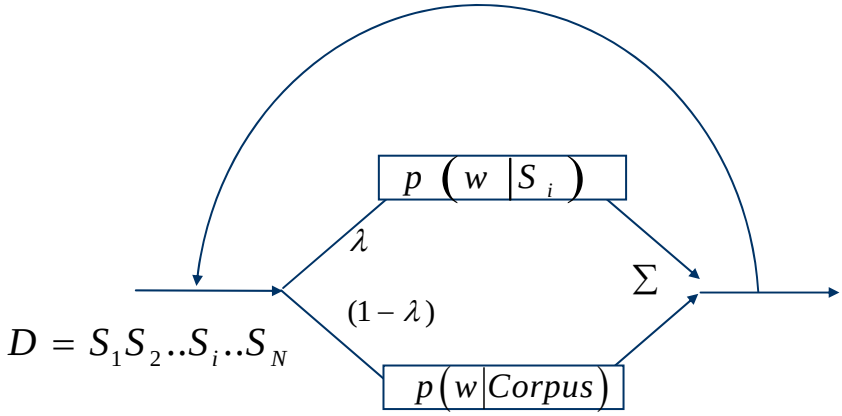


圖 3.3 隱藏式馬可夫模型-型二示意圖

同隱藏式馬可夫模型-型一的概念，當一篇文件進來時，視文件中每一字句為一機率生成模型，對於每個索引都有一個對應的機率分佈，文件中每一字句與文件的相關性，是藉由文件的所有索引在每一字句發生的相似值來決定，如圖 3.3 所示，數學式如下：

$$p(D|S_i) = \prod_{w \in D} p(w|S_i) \approx \prod_{w \in D} \left[\lambda p(w|S_i) + (1-\lambda) p(w|Corpus) \right] \quad (3.6)$$

其中 $p(w|S_i)$ 為文件中字句 S_i 產生索引 w 的機率值，並與一更大語料庫做平滑化， $p(w|Corpus)$。

演算法如下：

1. 將文件 D 斷句， $D = \{S_1, S_2, ..S_i.., S_N\}$
2. 對文件 D 中每一字句 S_i，計算其單連語言模型
3. 對文件 D 中各字句 S_i 估測 $p(D|S_i) \approx \prod_{w \in D} \left[\lambda p(w|S_i) + (1-\lambda) p(w|Corpus) \right]$ 機率值，並依此做排序形成一句排名
4. 依摘要比例，將句排名所對應的字句，輸出形成摘要

此外，對於每一個文件，將文件中每一字句 S_i 視為與文件 D 相關，則參數 λ 與每一字句 S_i 產生各索引 w 的機率值可藉由期望值最大化演算法 [Dempster *et al.* 1977]，自動調整參數與訓練模型，數學式如下所示：

$$\hat{\lambda} = \frac{\sum_{w \in D} E\left(w, S_i\right)}{|D|} \tag{3. 7}$$

$$\hat{p}(w \mid \mathrm{S}_i) = \frac{E\left(w, S_i\right)}{\sum_{w \in D} E\left(w, S_i\right)} \tag{3. 8}$$

$$E\left(w, S_i\right) = n\left(w, S_i\right) \cdot \frac{\lambda\, p\left(w \middle| S_i\right)}{\lambda\, p\left(w \middle| S_i\right) + (1-\lambda)\, p\left(w \middle| Corpus\right)} \tag{3. 9}$$

其中$|D|$是文件 D 的長度，$n\left(w, S_i\right)$是索引 w 出現在字句 S_i 的次數。

更進一步來說，因每個觀測（Observation）文件 D 中，皆含有模型 S_i 的資訊，是以可去除文件 D 中模型 S_i 的字詞，做字句移除（Sentence Removal），如下所示：

$$p(D \mid S_i) = \prod_{w \in D \wedge w \notin S_i} \left(\lambda p(w \mid S_i) + (1-\lambda) p\left(w \middle| Corpus\right)\right) \tag{3. 10}$$

3.4 主題混合模型（Topical Mixture Model, TMM）

根據 2.7 節關於主題混合模型的討論，給定一使用者查詢 Query $Q = q_1 q_2 .. q_n .. q_N$ ，一文件 D_i 可根據其相關程度做排名，$p(D_i \mid Q)$，經由推導後可由式(2.17)表示：

$$p(Q \mid D_i) = \prod_{n=1}^{N} \sum_{k=1}^{K} p(q_n \mid T_k)\, p(T_k \mid D_i)$$

延伸其應用於自動摘要模型上，將使用者查詢 Q 視查詢為一文件 D，一標題 H_i（標題可視為某一字句）可根據其相關程度做排名，$p(H_i \mid D)$，類同於 2.7 節的推導，最後可仿照式(2.17)表示成：

$$p(D \mid H_i) = \prod_{n=1}^{N} \sum_{k=1}^{K} p(q_n \mid T_k)\, p(T_k \mid H_i) \tag{3. 11}$$

也就是說，將原先以文件為模型，轉為以標題為模型。

於此以標題為模型主題混合模型中，可得到主題單連語言模型，如 $p(q_n \mid T_k)$，與其在各標題的權重，如 $p(T_k \mid H_i)$。

在訓練時，如果文件集已含有文件與標題相對應的資訊，如在一般新聞網站

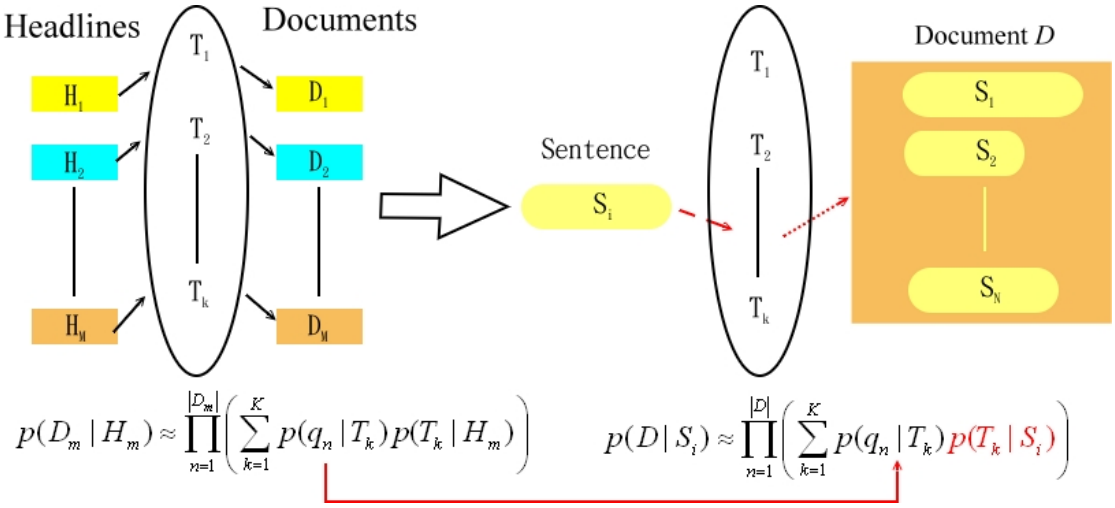


圖 3.4 主題混合模型示意圖

的新聞文件通常皆含有標題，則可藉由每一篇新聞的文件與其所相對應的標題來訓練。相對應的標題可使用當篇新聞的標題（本研究使用），也可由相關的新聞構成標題集，接著藉由式(2.20)-(2.22) 將 Q 轉為 D、D_i 轉成 H_i 做監督式訓練，以優化主題單連語言模型與其在各標題的權重。透過此訓練過程來學習標題（可視為字句）產生文件的流程。

在訓練時，如果文件集並無文件與標題相對應的資訊，則可將每一標題視為與自己相關，也就是將文件以標題取代，並藉由式(2.23)-(2.24) 將 Q 轉為 D、D_i 轉成 H_i 來進行非監督式訓練。

經由訓練過後，使用主題單連語言模型來代表主題的資訊。考慮如下情況，給定一使用者查詢文件 $D = q_1 q_2 .. q_n .. q_N$，文件中每一字句 S_i 可根據其相關程度做排名，$p(S_i \mid D)$，類同於 2.7 節的推導，最後可仿照式(2.17)表示成：

$$p(D \mid S_i) = \prod_{n=1}^{N}\sum_{k=1}^{K} p(q_n \mid T_k)p(T_k \mid S_i) \tag{3.12}$$

此機率，$p(D \mid S_i)$，即為主題混合模型在自動摘要的模型，其中主題單連語言模型由式(3.11)以標題為模型的主題混合模型訓練得之，如 $p(q_n \mid T_k)$，是以目前尚不知 $p(T_k \mid S_i)$ 的機率值，於此可利用原以標題為模型的主題混合模型，所得到的主題資訊，在摘要時即時迭代更新 $p(T_k \mid S_i)$，來估測每一字句 S_i 產生整篇文件 D 的機率，如圖 3.4 所示。

進一步來說， $p(T_k \mid S_i)$ 的初始值，可用下式估計：

$$p(T_k \mid S_i) = \frac{R(\overrightarrow{S_i}, \overrightarrow{T}_k)}{\sum_{r=1}^{k} R(\overrightarrow{S_i}, \overrightarrow{T}_r)} \tag{3. 13}$$

其中主題 T_k 是由原以標題為模型的主題混合模型而來，$R(\overrightarrow{T}_k, \overrightarrow{S_i})$ 代表利用餘弦估測字句 S_i 與主題 T_k 的距離，如下所示：

$$R(\overrightarrow{T}_k, \overrightarrow{S_i}) = \frac{\overrightarrow{T}_k \cdot \overrightarrow{S_i}}{\left\|\overrightarrow{T}_k\right\| \cdot \left\|\overrightarrow{S_i}\right\|} \tag{3. 14}$$

得到 $p(T_k \mid S_i)$ 的初始值之後， $p(T_k \mid S_i)$ 可藉由非監督式訓練，視每一字句 S_i 與自己相關，即時迭代更新得之，如下所示：

$$\hat{P}(T_k \mid S_i) = \frac{\sum_{q_s \in S_i} n(q_s, S_i)\, p(T_k \mid q_s, S_i)}{\left|S_i\right|} \tag{3. 15}$$

$$p(T_k \mid q_s, S_i) = \frac{p(T_k \mid S_i)\, p(q_s \mid T_k)}{\sum_{l=1}^{K} p(T_l \mid S_i)\, p(q_s \mid T_l)} \tag{3. 16}$$

$\left|S_i\right|$ 是字句 S_i 的長度， $n(q_s, S_i)$ 是查詢項 q_s 出現在字句 S_i 的次數， $p(T_k \mid q_s, S_i)$ 是在查詢項 q_s 與字句 S_i 出現的條件下潛藏主題 T_k 發生的機率。

在實作上，額外考慮每一查詢項在各字句中的重要性，是以式(3.12)可進一步延伸為：

$$p(D \mid S_i) = \prod_{n=1}^{N} \left(\lambda\, p(q_n \mid S_i) + (1-\lambda) \sum_{k=1}^{K} p(q_n \mid T_k)\, p(T_k \mid S_i) \right) \tag{3. 17}$$

$p(q_n \mid S_i)$ 為字句 S_i 產生查詢項 q_n 的機率， $p(q_n \mid T_k)$ 可由以標題為模型的主題混合模型訓練得之， $p(T_k \mid S_i)$ 可經由非監督式訓練即時迭代更新得之。

演算法如下：

1. 訓練以標題為模型的主題混合模型 $p(D \mid H_i) = \prod_{n=1}^{N} \sum_{k=1}^{K} p(q_n \mid T_k)\, p(T_k \mid H_i)$ ，得到主題單連語言模型用以代表潛藏主題的資訊

2. 將文件 D 斷句， $D = \{S_1, S_2, ..S_i.., S_N\}$

3. 由式(3.17) 估測 D 在每一字句 S_i 的機率值， $p(D|S_i)$：計算各字句 S_i 的單連語言模型，如 $p(q_n|S_i)$，與查詢項 q_n 發生在潛藏主題及字句產生各別主題的機率值，$\sum_{k=1}^{K} p(q_n|T_k)p(T_k|S_i)$。並依此機率值做排序，形成一句排名

4. 依摘要比例，將句排名所對應的字句，輸出形成摘要

此外，對於每一個文件，將文件中每一字句 S_i 視為與文件 D 相關，則參數 λ 與每一字句 S_i 產生各索引 w 的機率值可藉由期望值最大化演算法 [Dempster *et al.* 1977]，自動調整參數與訓練模型，數學式如下所示：

$$\hat{\lambda} = \frac{\sum_{w \in D} E(w, S_i)}{|D|} \tag{3. 18}$$

$$\hat{p}(w \mid S_i) = \frac{E(w, S_i)}{\sum_{w \in D} E(w, S_i)} \tag{3. 19}$$

$$E(w, S_i) = n(w, S_i) \cdot \frac{\lambda p(w|S_i)}{\lambda p(w|S_i) + (1-\lambda)\sum_{k=1}^{K} p(w|T_k)p(T_k|S_i)} \tag{3. 20}$$

其中 $|D|$ 是文件 D 的長度，$n(w, S_i)$ 是索引 w 出現在字句 S_i 的次數。

更進一步來說，因每個觀測（Observation）文件 D 中，皆含有模型 S_i 的資訊，是以可去除文件 D 中模型 S_i 的字詞，做字句移除（Sentence Removal），如下所示：

$$p(D|S_i) = \prod_{w \in D \wedge w \notin S_i} \left(\lambda p(w|S_i) + (1-\lambda)\sum_{k} p(w|T_k)p(T_k|S_i) \right) \tag{3. 21}$$