數位語音處理概論 – Final Project

資工三 B05902023 李澤諺

主題：Speech – Based Information Retrieval

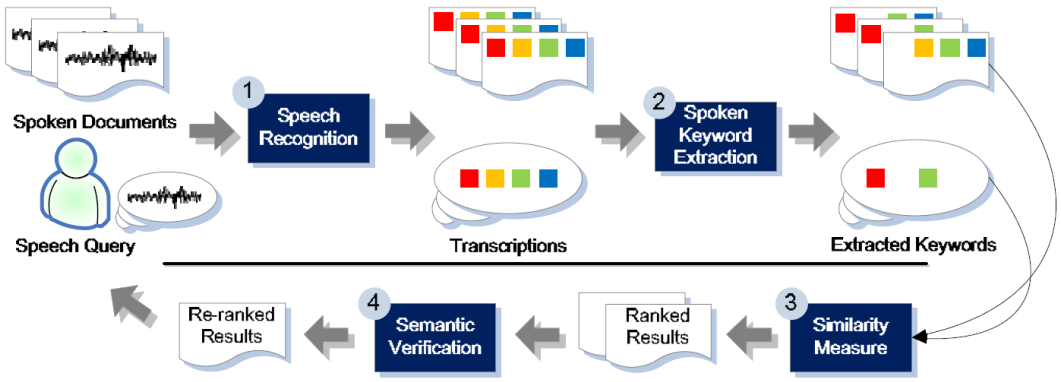
一、動機

在現代生活中，搜尋引擎已然成為科技領域中一個不可或缺的重要工具，其技術也正日新月異地蓬勃發展中，例如除了文字輸入外，現在也有可以使用語音作為輸入的搜尋引擎，為科技生活帶來了更多便利。此外，speech – based information retrieval 的概念與知識也有在課堂中介紹過，由此可以看出其重要性。因此在這份 final project 中，挑選了兩篇與 speech – based information retrieval 相關的論文、文獻，討論其中所用到的方法，以此更加深入了解與複習課程中所學到的知識，或得到課程以外的新知。

二、討論

[1] “Speech retrieval using spoken keyword extraction and semantic verification”

本篇論文的 retrieval system 為使用 speech query 搜尋 spoken document。其流程分為四個步驟，如下圖所示：



首先，speech recognition 的方法正如課程中期中考之前所學，作者先將音訊轉為 26 dimensional feature vector(12 MFCCs、12 delta MFCCs、1 delta – log energy、1 delta – delta log energy)再作辨識。由於作者為台灣人，因此作者使用了 150 個 Mandarin sub – syllable 建立 HMM，其中包含了 112 個 right – context – dependent INITIAL 和 38 個 context – independent FINAL，INITIAL 的 HMM 有 3 個 state，而 FINAL 的 HMM 有 4 個 state，每個 state 皆有 2 到 32 個 Gaussian，此外，作者也建立了一個 silence model，其為僅有 1 個 state 的 HMM，該 state 中有 64 個 Gaussian。而 language model 方面，作者使用了 MATBN(Mandarin Chinese broadcast news corpus)，其中包含了大約 3 年的新聞資料，每則新聞中平均約有 89.59 個字，以此建立了 language model。至此，作者建立了 HMM – based Mandarin LVCSR，用作 speech recognition。

接著，第二個步驟為 spoken keyword extraction，將 speech query 和

spoken document 中的 keyword 找出。作者為每一個 word w 計算一個分數：

$$A(w) = \lambda_S S(w) + \lambda_C C(w) + \lambda_L L(w)$$

其中 $S(w)$ 為 speech recognition confidence，$C(w)$ 為 prosody significance，$L(w)$ 為 word significance，而 λ_S、λ_C、λ_L 為 weight。

Spoken document retrieval 和 text document retrieval 最大的不同在於，spoken document 需要先作 speech recognition，其經常會發生 recognition error，使得 spoken document retrieval 比 text document retrieval 更為困難，因此，作者使用 speech recognition confidence $S(w)$ 衡量 w 被辨識正確的可能性，其為使用 generalized posterior probability 所計算出來，若 w 被辨識正確的機率 $S(w)$ 越高，其分數 $A(w)$ 也會越高。

此外，雖然 spoken document retrieval 比 text document retrieval 更為困難，但是 spoken document 可能可以提供比 text document 還要更多的訊息，例如語音訊息中會有 prosody，其可以反映 speaker 的情緒與目的，或是利用 speaker 的語速快慢、停頓間格、聲調與音量等等，來判斷何者為 keyword，作者利用這些特性為每個 w 計算其 prosody significance $C(w)$，若 $C(w)$ 越高，代表 w 在 speaker 的語音訊息中越被強調，就越有可能是 keyword，其分數 $A(w)$ 也會越高。其實 prosody significance 的概念在課堂中也有介紹過，我們可以設計一些利用 prosody 判斷語音訊息重要性的準則，每次選取不同的準則建立許多 decision tree，形成 random forest，用 random forest 幫助我們判斷該語音訊息的 prosody significance。

最後，word significance $L(w)$ 為使用 tf 和 idf 來計算：

$$L(w) = \frac{\text{freq}_w + 1}{\text{len}_d} \times \log \frac{N}{n + 1}$$

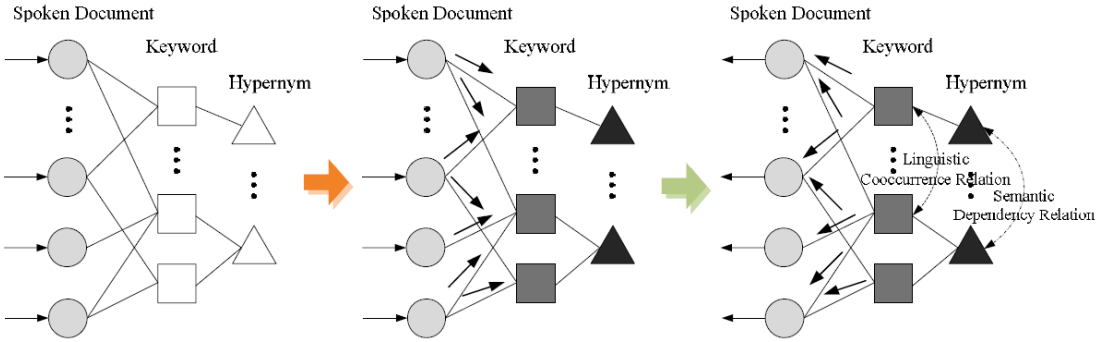
tf 和 idf 的概念在課程中有介紹過，因此在此不再詳細說明，但由此可以看出，只要 word significance $L(w)$ 越高，表示 w 越重要，越有可能為 keyword，其分數 $A(w)$ 也會越高。

利用以上方式，將 $A(w)$ 計算出來後，若 $A(w)$ 越高，w 就越有可能為 keyword，作者將 speech query 和 spoken document 中可能的 keyword 留下後，在第三個步驟的 similarity measure 中，利用 vector space model，將 speech query 和 spoken document 表示為 feature vector，分別記為 $\bar{q}$、$\bar{d}$，並用 cosine similarity 計算 $\bar{q}$ 和 $\bar{d}$ 的相關性：

$$\text{sim}(\bar{q}, \bar{d}) = \frac{\bar{q} \cdot \bar{d}}{\|\bar{q}\|\|\bar{d}\|} = \frac{\sum_{t=1}^{M} q_t \times d_t}{\sqrt{\sum_{t=1}^{M} q_t^2} \times \sqrt{\sum_{t=1}^{M} d_t^2}}$$

以此找出和 query 相關性可能較高的前 N_d 個 document。

到目前為止，前三個步驟的做法基本上皆有在課程中介紹過，而最後一個步驟 semantic verification，則是該篇論文中所獨有而重要的做法。由於 first retrieval 的正確率可能不高，因此通常需要 second retrieval，將 retrieved document 進行 re – rank，再將結果呈現給使用者，在課程中有介紹過 PRF 等方法，而在本篇論文中，則是使用 semantic verification，其流程如下圖：



首先，先將 N_d 個 retrieved document 中所有的 keyword 列出，並為每一個 keyword w_i 計算其 forward score $\alpha(w_i)$：

$$\alpha(w_i) = \sum_{d=1}^{N_d} \pi_d \times A'(w_i)$$

其中，$\pi_d = \text{sim}(\bar{q}, \bar{d})$，為 retrieved document d 和 query q 之間原始的相關性分數，而 $A'(w_i)$ 定義如下：

$$A'(w_i) = \lambda_S S(w_i) + \lambda_C C(w_i) + \lambda_I I(w_i)$$

$S(w_i)$ 和 $C(w_i)$ 的定義同前，而 $I(w_i)$ 為 w_i 在 retrieved document 中的 significance level，其定義為：

$$I(w_i) = \frac{freq_{w_i} + 1}{len_d} \times \log \frac{n+1}{N}$$

其定義和 L(w) 類似，差別在於 idf 中，若一個 word w 在許多 document 中皆有出現，則 w 的代表性不夠，其重要性可能不高，因此定義 $idf = \log \frac{N}{n+1}$，當 w 在越多 document 中皆有出現，其 idf 分數就會越低，但現在 w_i 為可能的 keyword，若其在 retrieved document 中出現的次數越多，表示該 keyword 越有可能為搜尋的目標，越為重要，因此在 $I(w_i)$ 的定義中是乘以 $\log \frac{n+1}{N}$，若 w_i 在 retrieved document 中出現的次數越多，則 $I(w_i)$ 越大。由此可以看出，若 keyword w_i 在 retrieved document d 中被辨識正確的機率 $S(w_i)$、prosody significance $C(w_i)$、以及 significance level $I(w_i)$ 越高，則代表 keyword w_i 在 retrieved document d 中來說相對重要，因此 $A'(w_i)$ 也會因此較高，而若 retrieved document d 和 speech query q 相關性越高，也會連帶表示 keyword w_i 在本次搜尋中更為重要，因此將 $A'(w_i)$ 對 π_d 作 weighted sum，以此得到 forward score $\alpha(w_i)$。

接著，將每個 keyword 的 semantic 或 hypernym 列出，並以此為每一個

retrieved document d 計算 backward semantic verification score β(d)：

$$\beta(d) = \sum_{i=1}^{N_K} \alpha(w_{i,d}) \times B(w_{i,d}) \times G(w_{i,d})$$

其中 N_k 為 d 中所有的 keyword 數，$w_{i,d}$ 為 d 中的第 i 個 keyword。由於不同字詞可以有著同一個語意，考慮到可能有些 keyword 語意相同，因此除了 keyword 出現的頻率會影響到其重要程度外，若其語意出現次數越高，或有越多字詞的語意相近，也能反映出該 keyword 的重要性，因此作者利用列出的 hypernym，以語意來計算每個 keyword 的重要程度 semantic dependency relation score $G(w_{i,d})$。另外，若不同字詞同時出現的機率越高，則也有可能表示這些字詞越重要(例如在課程中有提過 key phrase 的概念)，因此計算任意兩個 keyword 之間 bi – gram 的機率，並以此算出每個 keyword 的 bi – gram relation score $B(w_{i,d})$，在有了 $\alpha(w_{i,d})$、$B(w_{i,d})$、$G(w_{i,d})$ 之後，利用上面的公式為每個 retrieved document d 計算其 backward semantic verification score β(d)，若 d 中有著越多越重要的 keyword，表示 d 的重要性越高，因此將 retrieved document 用 β(d) 重新排序後，再將搜尋的結果呈現給使用者，正確率就有可能更高。

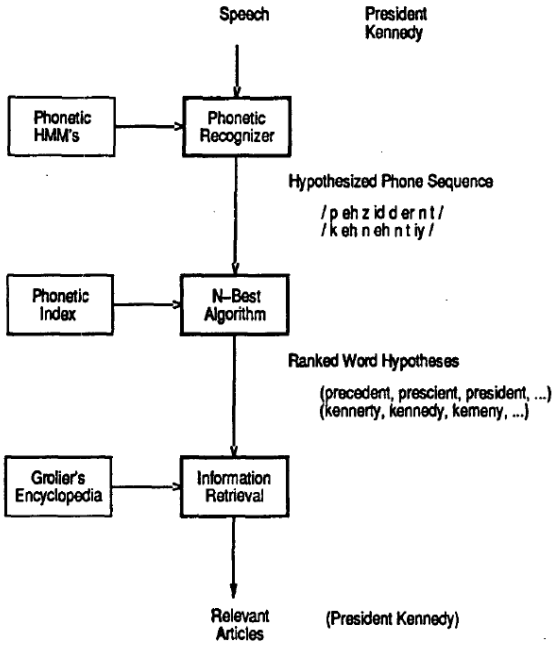
第四個步驟 semantic verification 雖然是本篇論文中獨有而重要的方法，但其基本想法和課程中提到的 semantic retrieval 相似：semantic retrieval 是找出和 query 的 semantic 相關的 document，而本篇論文中的 semantic verification 則是用 word 查找 document 後，再用 semantic 驗證 retrieved document 的重要性，類似用 semantic 進行 PRF。

以上四個步驟，即為本篇論文中在 speech – based information retrieval 所使用的方法。

[2] “Speech-Based Retrieval Using Semantic Co-Occurrence Filtering”

第二篇為更早期的文獻，其為使用 speech query 搜尋 text document。其流程分為三個步驟，如下頁圖中所示。

首先，第一個步驟為 speech recognition，和前一篇論文不同的地方在於，前一篇論文為在 speaker 輸入一整段音訊後，再進行 LVCSR，而本篇文獻則是要求 speaker 將每個 word 依次輸入，用以進行 isolated word recognition，此外，本篇文獻中的 speech recognizer 為 speaker dependent：先使用 TIMIT speech database 訓練好的 speaker independent HMM model 作為 initial model，其中，每個 HMM 皆有 3 個 state，而且所有 Gaussian 的 covariance matrix 皆為 diagonal matrix，以此作為 initial model，再將該 speaker 說出的 1000 個 isolated word 轉為 feature vector(14 Mel – scaled cepstra 與其 derivative、1 log energy derivative)，用其將 initial model 調整為 speaker dependent model，文中並未提到其調整方法，但我們在課堂中已經有學過很多技術，例如 MAP、MLLR、eigenvoice 等等，皆可用於 speaker adaptation。有



了 HMM model 後，在 speaker 輸入一個 word 要做 speech recognition 時，便可使用 Viterbi algorithm 找出其可能的 phone sequence，再用事先建立好的 lexicon(文中稱為 phonetic dictionary，其中有 175000 個 word，並為每個 word 標上其 phonetic spelling，包含 phonological variant、alternative pronunciation 等等，若有新加入的 word 不知道其 phonetic spelling，則用 text – to – speech synthesis，自動生成其可能的 phonetic spelling)，以此比對出所有可能的 phone sequence 其代表的 word。

接著，第二個步驟為 n – best matching。由於 speech recognition 經常會有 recognition error，因此在前一個步驟中辨識機率最高的 word，不一定為 speaker 真正想輸入的 word，故作者使用了以下兩個方法 generate and test 和 HMM search 其中之一，列出最有可能為 speaker 想要輸入的 n 個 word。

在 generate and test 和 HMM search 中，皆要先計算第一個步驟中所列出的所有 phonetic sequence，其發生 insertion、deletion 和 substitution 的機率，方法為先將每個 phonetic sequence 和其代表的 word 在 phonetic vocabulary 中正確的 phonetic sequence 做 alignment(在課程中有介紹過如 DTW，可以將兩個音訊做 alignment)，再用 Laplacian estimation 求出 insertion、deletion 和 substitution 的機率。

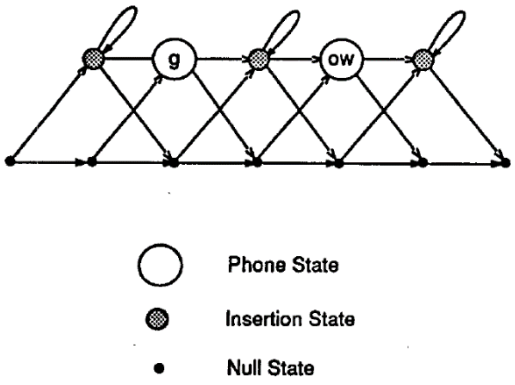
有了以上的機率後，即可進行 generate and test 或 HMM search。由於作者發現使用 HMM search 的正確率比 generate and test 高，因此在文章中對 generate and test 的介紹較少，而較著重於 HMM search。Generate and test 的方法為，將第一個步驟中所有可能的 phonetic sequence 依照其被辨識出來的機率由高到低做排序，接著從辨識機率最高的 phonetic sequence 開始，依照 insertion、deletion 和 substitution 的機率，將其還原成發生 insertion、deletion 和 substitution 前最有可能的原 phonetic sequence，若還原的結果存

在於 phonetic vocabulary 中，便將其加入 n – best hypothesis 中。

而 HMM search 的作法為，考慮在 recognizer 辨識出的 phonetic sequence 為 $y_1y_2\cdots y_p$ 的條件下，speaker 真正想輸入的 word 為 $w = w_1w_2\cdots w_q$ 的機率為：

$$p(w \mid y_1y_2\cdots y_p) = \frac{p(w)p(y_1y_2\cdots y_p \mid w)}{p(y_1y_2\cdots y_p)}$$

若 $p(w)$ 為 uniform，則根據 MAP principle，當 $p(y_1y_2\cdots y_p \mid w)$ 有最大值時，便可以使得 $p(w \mid y_1y_2\cdots y_p)$ 有最大值，因此 n – best hypothesis 中，選取可以使得 $p(y_1y_2\cdots y_p \mid w)$ 最大的前 n 個 phonetic sequence 所代表的 word。而找出這 n 個 phonetic sequence 的方法為，以 speaker 想輸入的 word w 為"go"，其正確的 phonetic sequence 為/g/ow/為例，建立以下的 HMM：



其中 phone state 代表正確的 phonetic sequence 中所有的 phone，insertion state 代表有哪些 phone 可能會在 speech recognition 被 insert 到 phone state 之間，而 null state 則是用於使得 phone state 在 speech recognition 時可以產生 deletion，而不同 state 之間的 transition probability 為先前所計算出的 insertion 或 deletion 機率，每個 state 的 observation probability 則為先前所計算出的 substitution 機率。建立該 HMM 後，即可用 basic problem 2 的 backward algorithm，求出最有可能產生出 w 的前 n 個 phonetic sequence，作為 n – best hypothesis。HMM 這個數學模型可以用於許多不同領域，而在數位語音處理的領域中，除了作為 acoustic model 之外，也有 HMM search，找出 n – best hypothesis 如此應用，由此可以看出 HMM 的基本性、重要性與廣泛性。

最後，在為 speaker 輸入的每個 word 皆找出其 n – best hypothesis 後，第三個步驟即為 information retrieval。由於對於每一個 word，仍然不確定其 n – best hypothesis 中何者為 speaker 真正想輸入的 word，因此作者使用了 semantic co – occurrence filter，舉例來說，若 speaker 想輸入的 word 分別為"president"和"Kennedy"，而其 n – best hypothesis 分別為：

president: (precedent, prescient, president...)
kennedy: (kennerty, kennedy, kemeny, remedy...)

其中，"president"和"Kennedy"皆不為其 n – best hypothesis 中的第 1 名，但考

慮到若不同的 word 語意相近，或不同的 word 在 document 中同時出現的機率很高，甚至組合在一起時會形成 keyphrase 等等情況，則將不同 n – best hypothesis 中的 word 作組合後，才能真正反映出這些 word 的重要性，因此作者將不同 n – best hypothesis 中的 word 作組合後，才將這些組合輸入 retrieval system 作搜尋，以上述例子來說明，作者可以組合出 {precedent kennerty , precedent kennedy , prescient kennerty , predcient kennedy , president kennerty , president kennedy , … }，其可以用以下的 Boolean operation 來表示：

(AND 15 (OR precedent, prescient, president, resident...)
(OR kennerty, kennedy, kemeny, remedy...))

此即為 semantic co – occurrence filter。而在將這些組合輸入 retrieval system 搜尋時，搜尋的方法為使用 Boolean retrieval，Boolean retrieval 的作法為，先將所有 document 中所有的 word 列出，再為每一個 word 紀錄其出現在哪一些 document 之中，如此一來，在輸入 query 之後，例如 query 包含的 word 為”president”和”Kennedy”，則只要分別找出”president”和”Kennedy”各自出現在哪一些 document 之中，再取這些 document 的交集，即為搜尋的結果，由於以上步驟皆可使用 Boolean operation 來達成，因此稱為 Boolean retrieval。至此，即為該文獻中 speech – based information retrieval 的作法。

三、結語

事實上這兩篇論文、文獻距離現今皆已經有很長一段時間了，在科技迅速發展的現代，已經有了其它更多更好的方法與技術，讓 information retrieval 變得更為方便。在決定 final project 要討論這兩篇論文、文獻之前，我看了其它論文中所使用的各種方法，雖然不是每篇論文都看得懂其作法，但仍讓我對科技的發展與資訊領域中技術的多樣性更為驚嘆，不禁讓我自勉，若要在現今的科技生活與資訊領域中發展與研究，必須要多多充實自身、吸收新知，才能讓自己跟得上科技的腳步。此外，雖然各篇論文中皆有不同的技術，但其所使用到的基本技術或概念事實上皆大致相同，例如在 speech recognition 中，幾乎所有的論文皆使用了 HMM，而這些基本技術與概念皆有在課程中介紹過，由此可知課程中所介紹的概念是多麼的基本與重要，因此，未來在學習數位語音處理的路上，除了多多了解、補充新知外，也要記得本學期學到的這些基本而重要的技術，溫故而知新，才能在數位語音處理這條路上走得更久更遠。

四、參考資料

[1] “Speech retrieval using spoken keyword extraction and semantic verification”, IEEE, 2007

[2] “Speech-Based Retrieval Using Semantic Co-Occurrence Filtering”, published in HUMAN LANGUAGE TECHNOLOGY: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994