

Overview of conventional TTS

Sian-Yi Chen

Advisors : Tay-Jyi Lin and Chingwei Yeh

Outline

- 傳統 TTS 合成架構
 - Context-dependent Label
 - Conversion Process of Text-to-Label
 - Letter to Sound Rules
- HMM訓練架構
 - Hidden Markov Model (HMM)介紹
 - 訓練HMM模型
- 單音素HMM
- 三音素HMM
- 利用決策樹取得最佳參數
- 使用濾波器進行合成

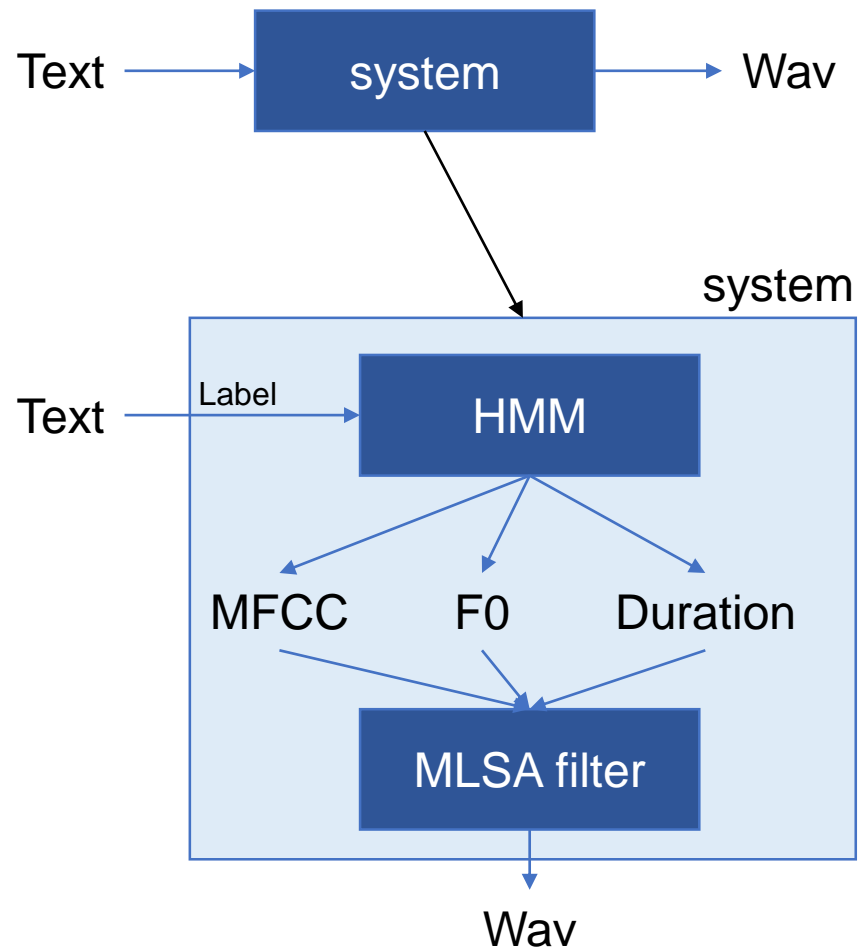
■ 傳統 TTS 合成架構

傳統 TTS：statistical parametric synthesis 統計參數合成 [1]

TTS (text-to-speech) 目標為輸入文字，輸出音檔

系統中主要使用 HMM (Hidden Markov Model) 預測出 3 個參數，並將這些參數使用 MLSA 濾波器合成語音。

- MFCC：梅爾倒頻譜係數，一種用於描述人類聲道形狀的特徵。
- F0：基頻，一發聲體發出聲音的時候，聲音可以分解成許多單純的正弦波，其中頻率最低的就是基音，其他較高的頻率都叫做泛音，主要用於描述音高、音色。
- Duration：決定一單位(音素)的時間長度。



Context-dependent Label

上下文標籤：輸入一段文字，解析語句中的音素、音節、詞性...等關係。

以第一個音素(ao)為例，標籤會將前音素 - 當前音素 + 後音素列為一組標籤 (sil-**ao**+**th**)，其中sil為silence，後面緊接解析出句子的關係，關係如下：

- 音素：以當前音素為主的前後音素是什麼？他們在音節中的什麼位置？...等
- 音節：該音節是否有包含重音？包含幾個音素？...等
- 詞性：

根據個別包含三個音素的單詞的詞性、在句中的位置...等

標籤格式：音素之間的關係、音節、詞性...等

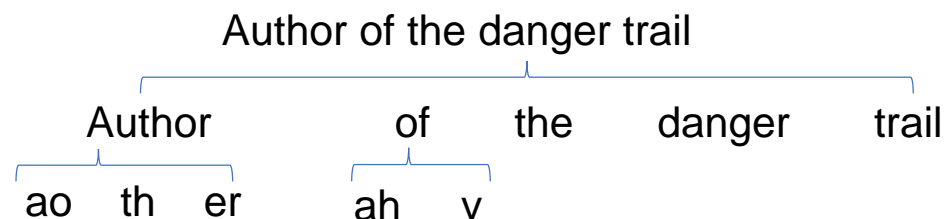
第一個標籤：sil-**ao**+**th** ...
第二個標籤：ao-**th**+**er** ...
第三個標籤：th-**er**+**ah** ...
⋮
⋮
⋮

以Author為例的上下文音素標籤

句子

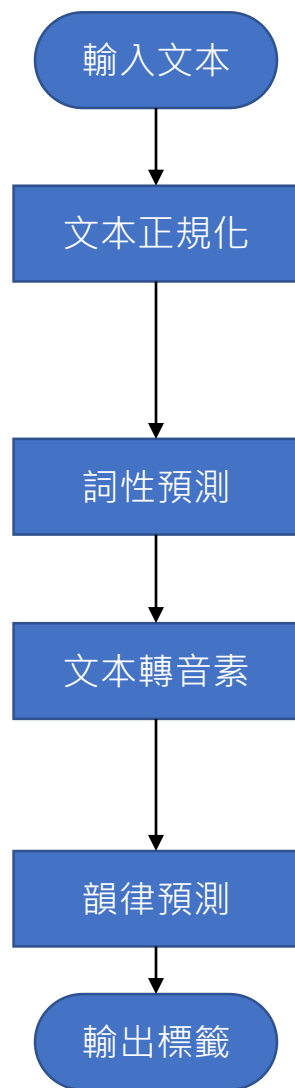
單詞

音素



句子拆分至音素

Conversion Process of Text-to-Label



通常為一句話，ex：Author of the danger trail

- 清除 Punctuation：刪除標點符號或特殊符號
- Tokenization (標記化)：可以透過正規表達式將一句話拆分成單詞
- 刪除 stop words：刪除功能/停用詞 (虛詞)，像是is、the、at、on...等
- Lemmatize/ Stemming：詞形還原/詞幹提取，如右
- Other：刪除網址、HTML標籤、表情符號...等

Lemmatize：將以下兩個字還原成原本的樣子amuse

- amused
- amusing

Stemming：以下四個字取出詞幹univers

- university
- universal
- universities
- universe

Lemmatize/ Stemming：詞形還原/詞幹提取

使用HMM標記詞性 [2]

使用字典查詢單詞對應到的音素，通常字典中包含三個部分 [2]

- addenda (由手動添加單詞組成)
- compiled lexicon (已預先編譯好的字典)
- 使用letter to sound rules (字母到聲音的規則) 處理上述2個都查找不到單詞的情況

使用ToBI標記韻律 [2]：透過定義各種基頻 (F0) 的聲調來標記韻律，如右下圖所示

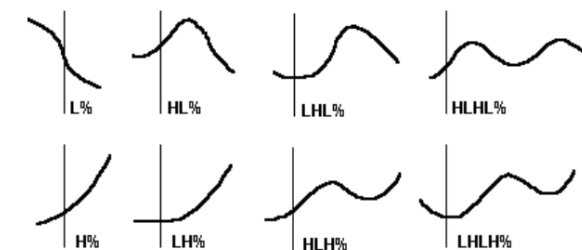


圖 3. IP 的八個邊界音的 f0 輪廓示意圖。

Letter to Sound Rules

在單詞轉換到音素的過程中，無法保證將字典包括所有單詞，因此使用Letter to Sound Rules (LTS) 規則來解決這個問題。

LTS模型建立步驟 [3]

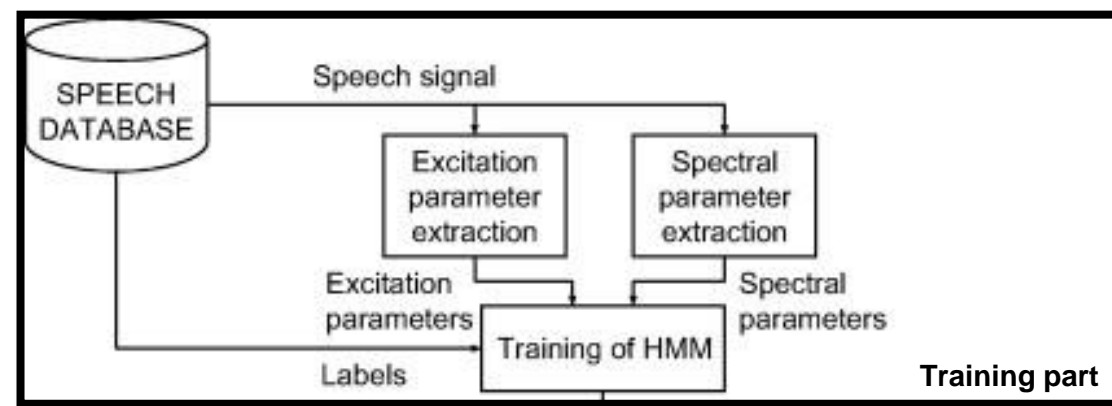
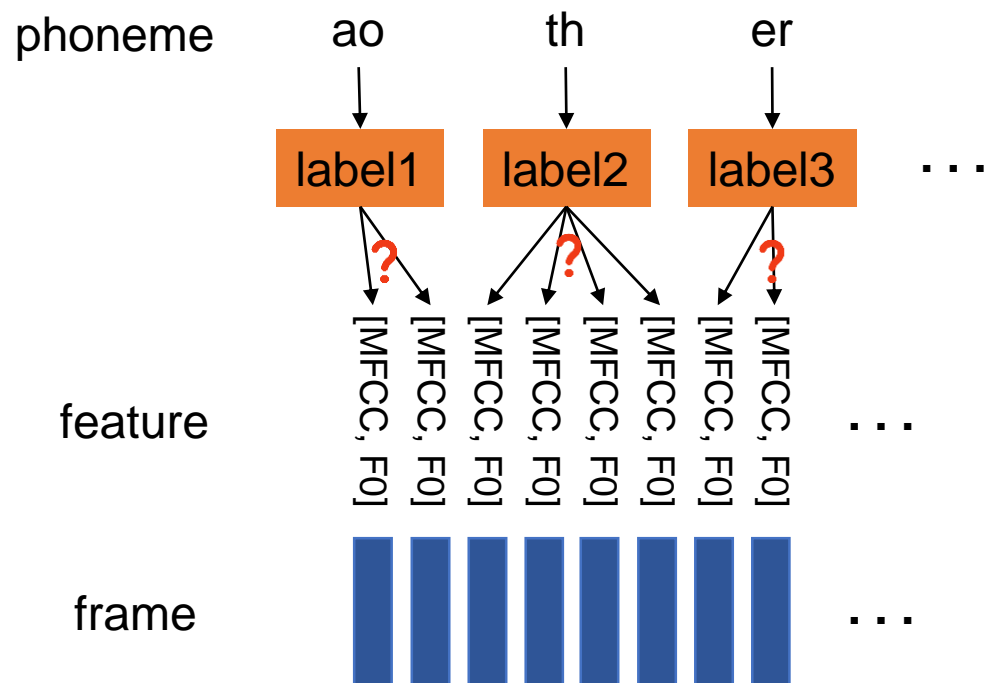
1. 選擇字典並預先處理成合適的訓練集
 - 刪除四個字母以下的短詞，因為這些單詞通常是功能詞(虛詞)，並很常包含縮寫，縮寫通常與它的發音不會有關係
 - 刪除所有非內容詞，因為要預測的詞通常不是常見的詞，並且他們通常具有標準的發音。
 - 將字典中每10個單詞就拆分成訓練和測試資料。
2. 定義一組允許的字母與音素配對
 - 一個單詞中的字母數量和發音中的音素數量通常不是一對一的配對，可能0個、1個，甚至多個，因此加入了epsilon來幫助對齊。
 - “_epsilon_: Almost all letters can in some context correspond to no phone.”
3. 建立每個字母/音素對的機率
 - 標記初訓練集中所有字母與音素的對齊方式，並計算所有對應的機率。
 - 若以C這個字母為例，列出所有可能的發音方式，C：_epsilon_ k ch s sh t-s
4. 將字母與一組相等的音素/_epsilon_ 對齊
 - 利用DTW將字母與音素做初始化對齊，並在最有可能的位置插入epsilon。
 - 做完初始化對齊就利用EM (maximization) 演算法進行迭代直至收斂，例如在CMU字典上需要5次迭代。
5. 構建用於根據字母 (和上下文) 預測電話的 CART 模型
 - 給定上下文 (如每邊3個) 的情況下訓練了CART樹。

[3] A. W. Black, K. Lenzo and V. Pagel, “Issues in building general letter to sound rules,” in *Proc. ISCA SSW*, 1998.

HMM訓練流程與目的

Database中包含語音與文本，其中語音用於提取聲學特徵 (MFCC與F0)，而文本則轉換成包含上下文內容的標籤。

目標是將標籤標記於聲學特徵上，並透過訓練HMM預測模型得到一個標籤對應到幾個聲學特徵。



HMM訓練流程

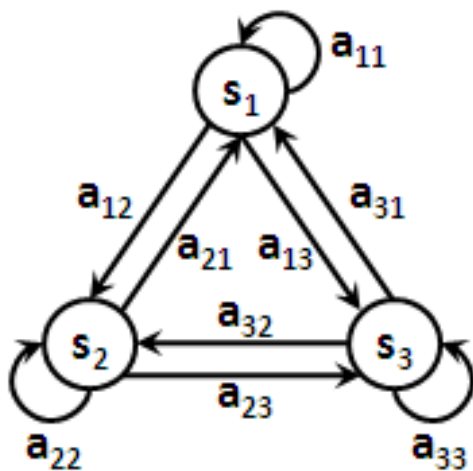
Hidden Markov Model (HMM)

Markov Model：選一個狀態作為起點，然後沿著邊隨意走訪任何一個狀態，會一直走並沿途累計從起點該點的機率。

$$S (\text{狀態}) = \{S_1, S_2, S_3\}$$

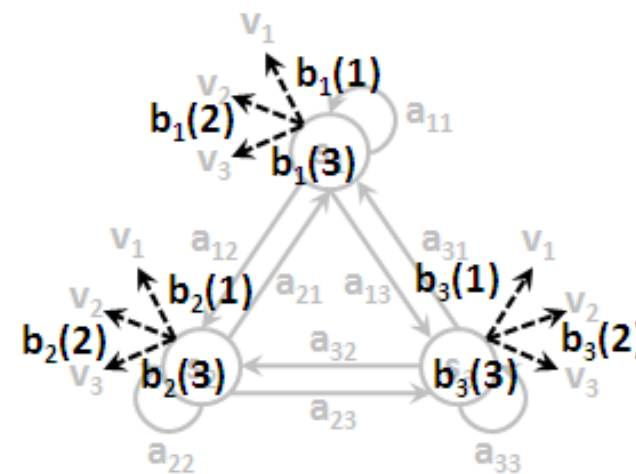
$$A (\text{轉移機率}) = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$$

Π (起始機率) = 可以取任一點作為起點，機率總和為1



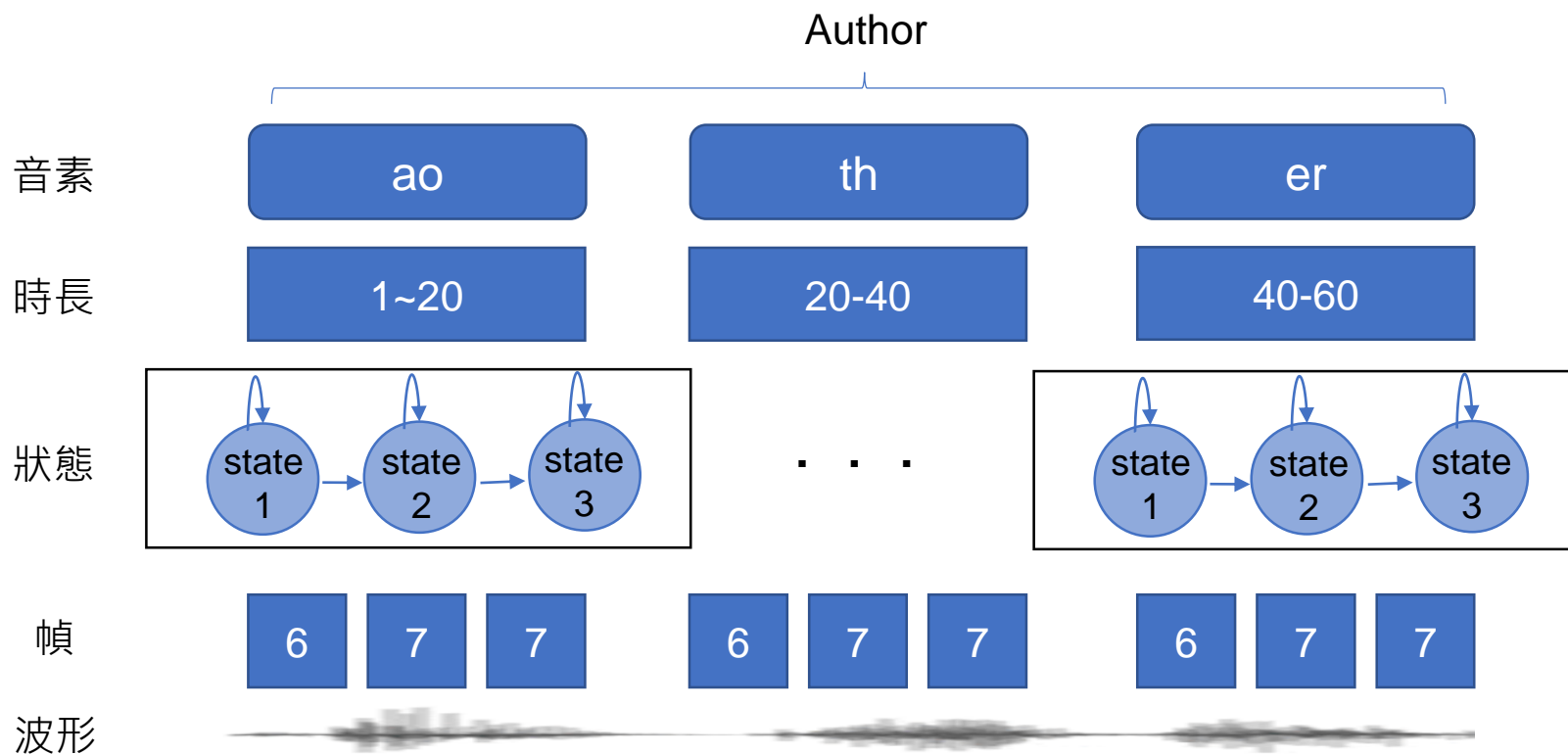
Hidden Markov Model：隱藏馬可夫模型添加了新的要素，每造訪一個狀態就會出現一個新的值 (v)，而每一個新出現的值都有不同的機率 (b)。

舉例來說今天有一位醫生要判斷病人是健康的還是發燒，病人只會回答正常(S_1)、頭暈(S_2)、冷(S_3)，醫生要從這3個答案中判斷是否發燒，是否發燒就是隱藏狀態(無法直接觀察到)；發燒(v)的機率(b)。

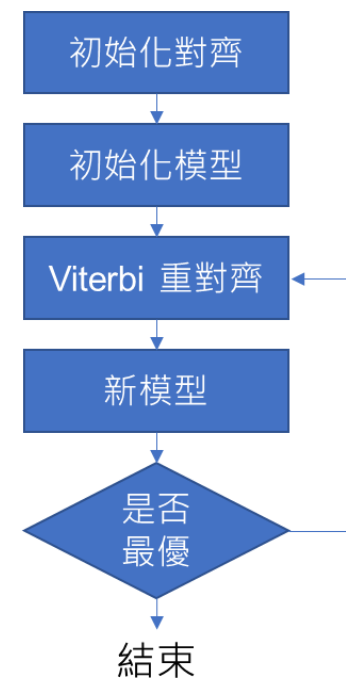


HMM模型初始化對齊

單詞可以透過CMU發聲字典轉換成音素，但一開始不知道一段語音的哪些幀對應哪些狀態，因此進行**初始化對齊**，也就是將一段時長平均分配，假設 **author** 這個詞發聲 1.5 秒，若一個 **frame** 長 25ms，一次移動 25ms，則可以得到 60 個**frame**，也就是“ao”、“th”、“er” 每個音素各對應至 20 個 **frame**，每個音素又由 3 個狀態所組成，因此每個狀態分配到 6 或 7 個 **frame**。



音素對齊frame示意圖



HMM 訓練流程

HMM 初始化模型

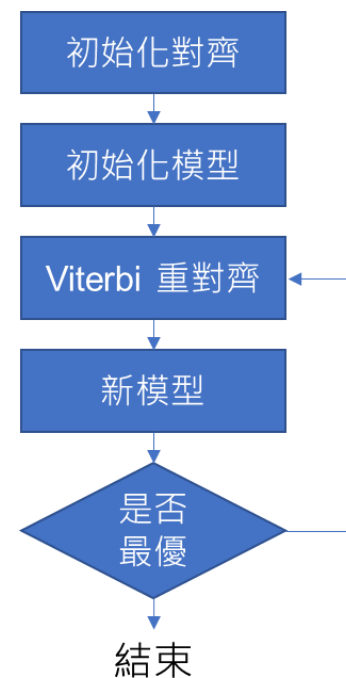
輸入：觀察序列 $O=[o_1, o_2, \dots, o_x]$ (X個frame的MFCC特徵)

輸出：通過模型計算每一 frame 對於“ao”這個音素的某一狀態 (3狀態) 的機率

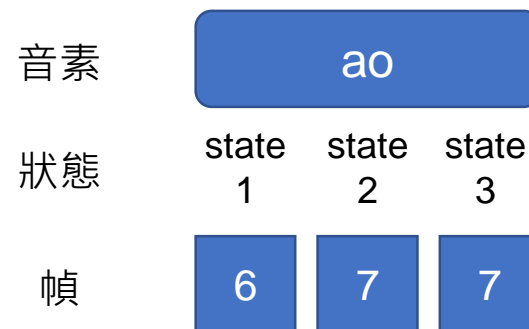
HMM模型 $\lambda=(A,B,\Pi)$

- A是狀態轉移機率的矩陣，決定狀態序列
- B是觀測狀態生成機率的矩陣，決定觀察序列
- Π 是隱藏狀態的初始機率分佈

- 初始化完就可得到轉移機率 A，計算轉移次數 (狀態1->狀態1，狀態1->狀態2)，轉移次數/總轉移次數 = 轉移機率
- 初始機率分佈 Π ：HMM 模型是從左到右的模型，一開始在狀態1的機率為100%，所以此參數可忽略
- 狀態生成機率 B：一個狀態對應一個HMM模型，一個狀態又對應好幾個frame，所以好幾個frame對應一個HMM模型，初始化後，可得知狀態1對應6個frame，因此可以透過此計算狀態1的HMM模型 (單高斯模型)，求得平均值和變異數。



HMM 訓練流程



音素對應至frame

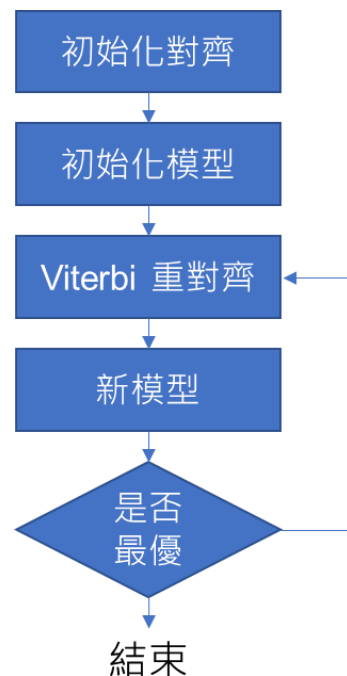
HMM 新模型

重新對齊

- 初始化結束需要重新對齊，使用viterbi動態規劃演算法，用於尋找最有可能產生觀測事件序列的維特比路徑(隱含狀態序列)。
- 根據初始化模型來計算，記錄每個時刻與每個可能狀態之間的最優路徑機率，並同時記錄最優路徑的前一個狀態，不斷向後反覆運算，找出最後一個時間點的最大機率值對應的狀態。

反覆運算

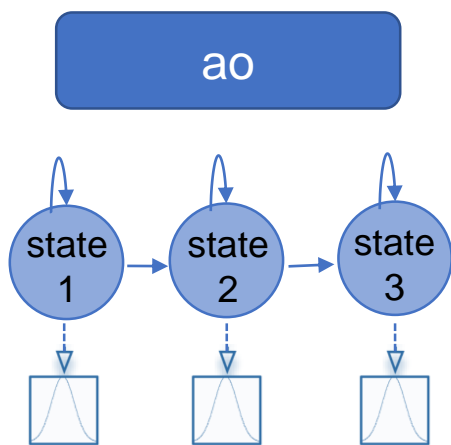
- 透過重新對齊可以得到新的轉移機率和生成機率，就可以進行下一次的對齊，尋找新的最優路徑，得到新的對齊，新的對齊繼續改變轉移機率和生成機率。
- 如此迴圈反覆運算直到收斂，則HMM模型訓練完成。
- 反覆運算次數可以透過設定固定的迴圈數，也可以藉由觀察似然 (某件事發生的機率) 的變化，如果變化不大就結束。



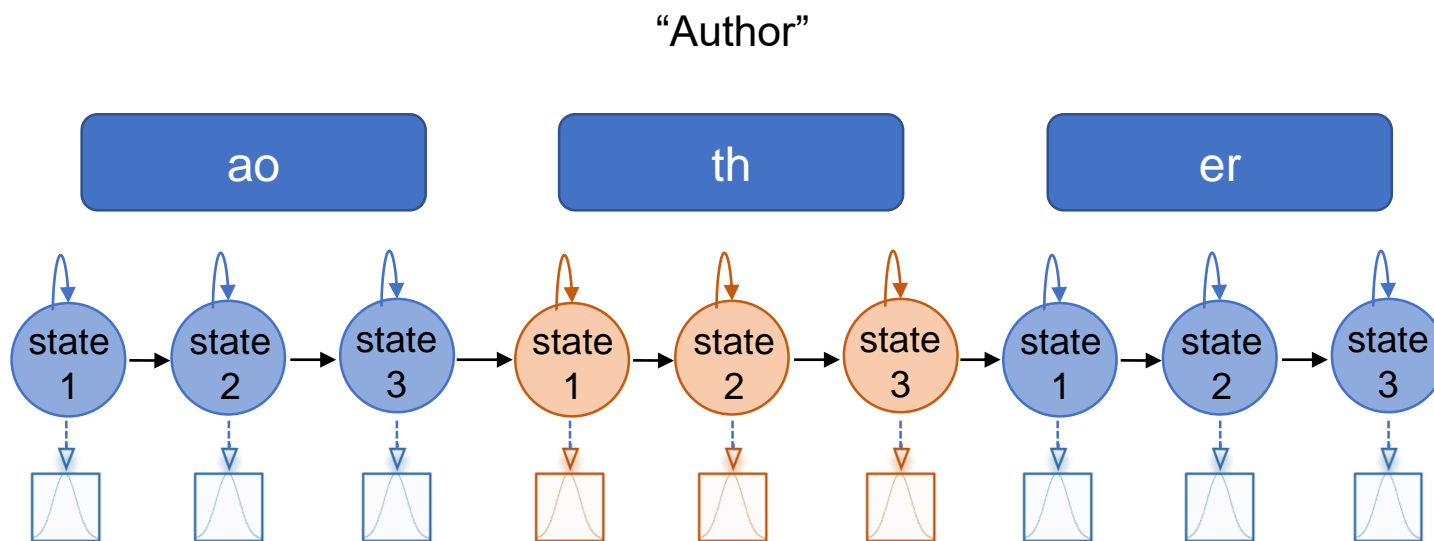
HMM 訓練流程

Utterance HMM

給定一個單詞並透過CMU發聲字典轉換成音素，然後將每個音素的HMM拼接起來就可以得到這個詞的HMM，同理，我們將很多詞的HMM連接起來就可以得到一個句子的HMM。
對於每個音素，通常使用三個狀態的HMM去建模，而三個狀態分別為起始音、持續音、結束音。



音素ao的HMM模型

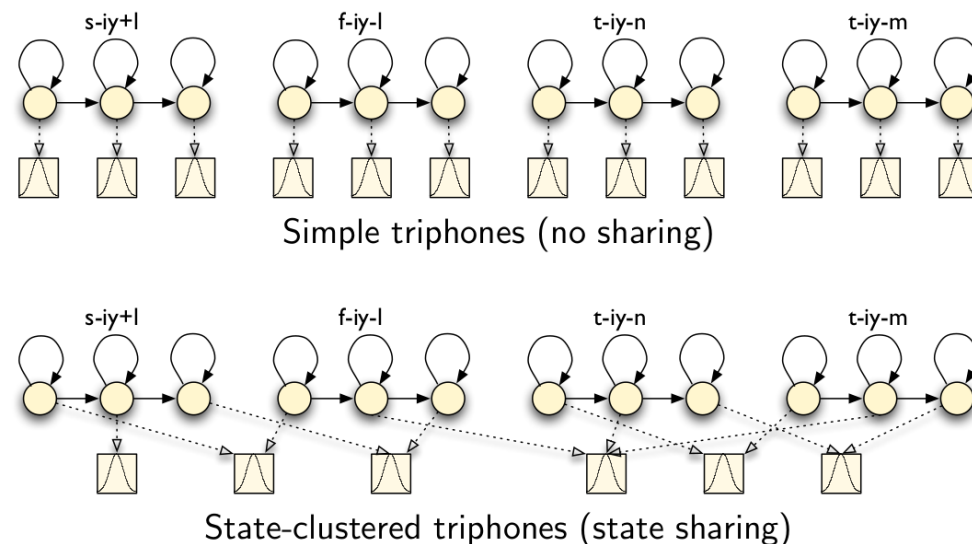
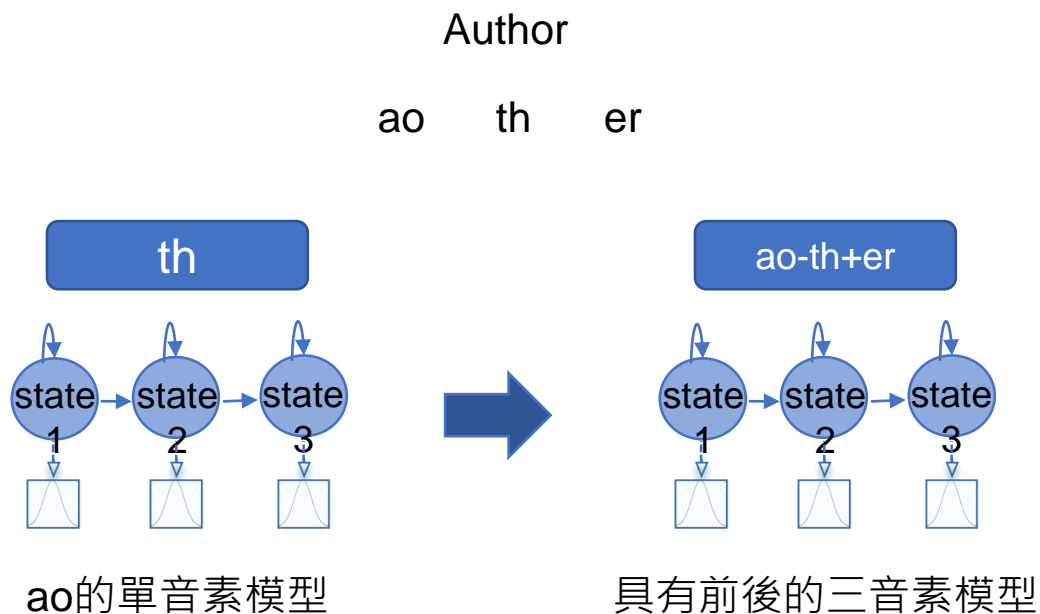


Author的HMM模型

三音素 HMM 模型

一句話或是一個詞中的音素，常常不是獨立發音，而是會隨著前後音素的改變也跟著改變發音。因此需要考慮上下文進行建模，一般考慮前、中、後的音素為一組。

假設現有3個音素，從單音素模型變成三音素模型時，HMM模型數量會成指數性成長，因此使用狀態綁定的方法來解決這個問題，狀態綁定就是讓具有相似特徵的一些狀態共用同一組模型參數，這樣就可以有效減少模型參數的數量。



■ 使用問題集進行聚類

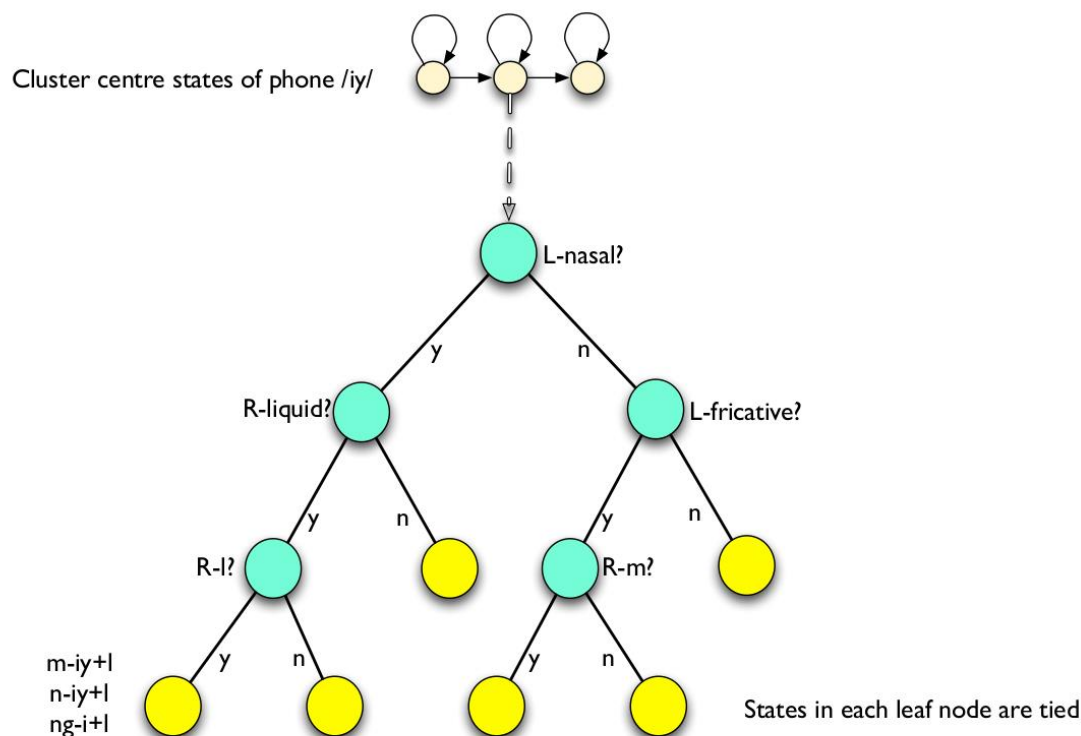
讓具有相似特徵的一些狀態共用同一組模型參數就稱為聚類。

通常使用語音決策樹(Phonetic Decision Tree)演算法，這是一種由上往下(分裂)，並不斷的分為兩類。

剛開始同一個音素的所有triphone都在根節點，然後每個節點都會根據問題進行分類，常見的問題如下：

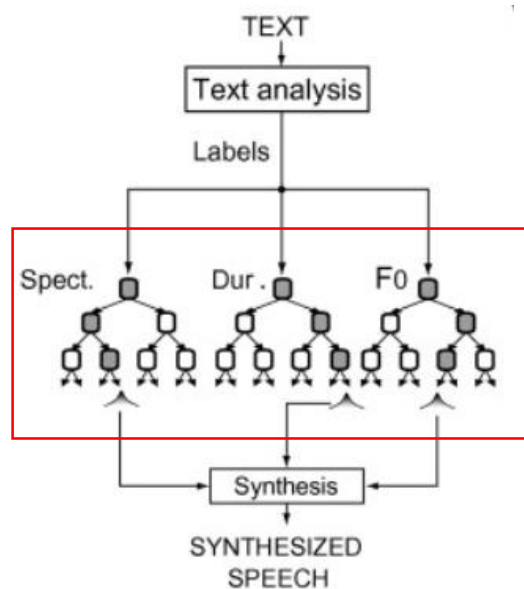
- 左邊是不是一個鼻音？
- 右邊是不是摩擦音？

目標是找到一個問題將一大堆HMM模型分成兩類。

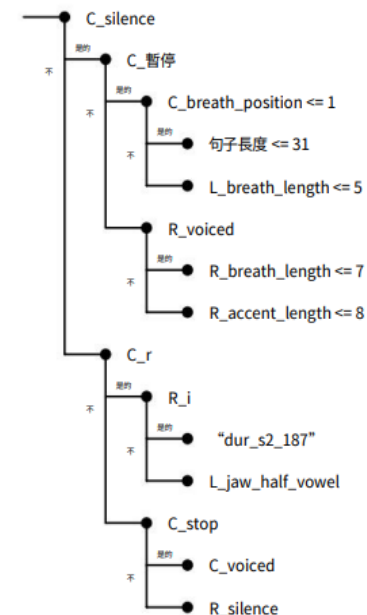
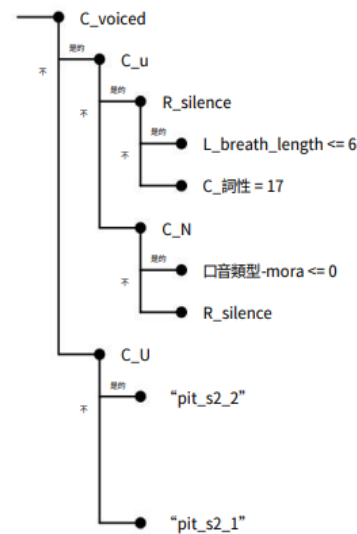
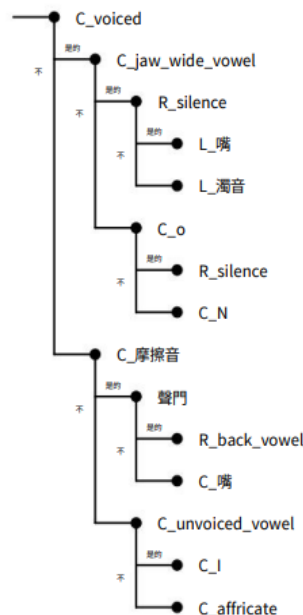


■ 頻譜、基頻、持續時間各別使用決策樹聚類

訓練完三音素模型，會得到很多HMM模型，因為三個係數(MFCC、F0、持續時間)都有各自的上下文關係，因此我們個別使用決策樹進行聚類，最後得到以下的三個樹，並可以得到三個參數的最優解。



(a) Statistical parametric synthesis



(a) Tree for Spectrum Model (b) Tree for Pitch Model (c) Tree for State Duration Model

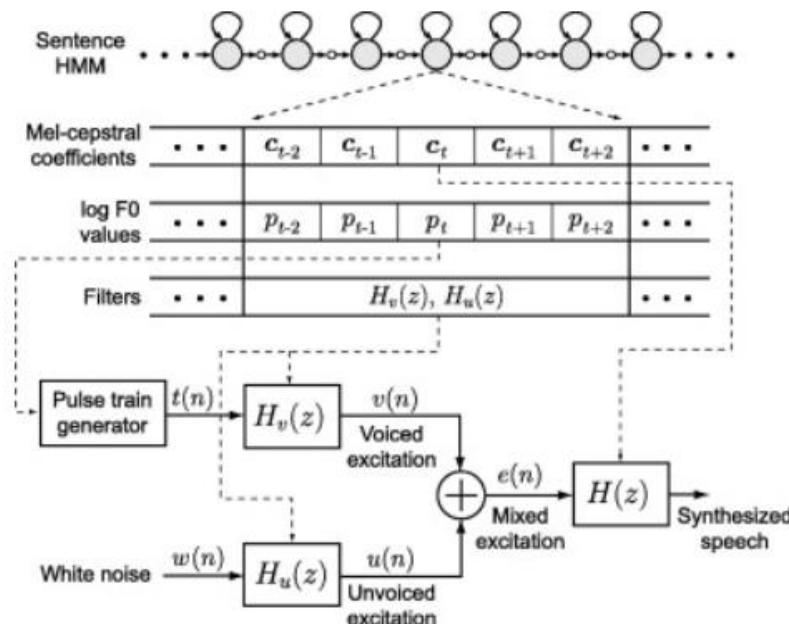
Examples of decision trees [2]

■ 使用濾波器進行合成

最後根據決策樹各別取得三種最佳參數，並輸入濾波器進行合成。

目的為將參數透過濾波器轉換回聲音，它透過簡單週期的脈衝序列或是白噪聲激勵通過濾波器達成轉換。

- 單週期脈衝序列對應到濁音，也就是人類聲帶振動發出的聲音。
- 白噪聲對應到清音，聲帶不震動發出的氣音。



HMM-based excitation scheme [1]

附錄

■ 上下文標籤格式

完整上下文標籤：

sil^sil-ao+th=er@1_2/A:0_0_0/B:1-1-2@1-2&1-7#1-4\$1-3!0-2;0-

4|ao/C:0+0+1/D:0_0/E:content+2@1+5&1+2#0+3/F:in_1/G:0_0/H:7=5@1=2|L-L%/I:7=3/J:14+8-2

所有特殊位元皆為固定格式的連接用符號，無意義

可以看到除了 p1~p7 總共分為 9 種關係，各類所代表的物理意義可參照下一頁每一種顏色對應其關係格式：

1. p1^p2-p3+p4=p5@p6_p7
2. /A:a1_a2
3. /B:b1-b2@b3-b4&b5-b6#b7-b8!b9-b10|b11
4. /C:c1+c2
5. /D:d1_d2
6. /E:e1+e2@e3+e4
7. /F:f1_f2 /G:g1_g2
8. /H:h1=h2@h3=h4
9. /I:i1_i2
10. /J: j1+ j2- j3

■ 生成上下文標籤

範例句子：Author of the danger trail

Author
ao th er

第一個標籤

持續時間 $x^x \cdot \text{sil} + \text{sil} = \text{ao}$ 其餘資訊 狀態一
開頭沒有聲音，sil(靜音)

第二個標籤

持續時間 $x^x \cdot \text{sil} + \text{sil} = \text{ao}$ 其餘資訊 狀態二

⋮

第六個標籤

持續時間 $\text{sil}^x \cdot \text{sil} \cdot \text{ao} + \text{th} = \text{er}$ 其餘資訊
前前音素 前音素 當前音素 下一個音素 下下一個音素

標籤格式(音素、音節、單詞語、短語、句子之間的關係)

2050000 2400000 sil^sil-ao+th=er@1 2/A:0 0 0/B:1-1-2@1-2&1-7#1-4\$1-3!0-2:0-4|ac/C:0+0+1/D:0_0/E:content+2@1+5&1+2#0+3/F:in_1/G:0_0/H:7=5@1=2|L-L%/I:7=3/J:14+8-2

前後音素、該音素在音節中的位置

前一個音節是否為重音、音素數量

當前音節重音、音素數量、在單詞中的位置、在短語中的位置...等

下一個音節是否為重音、音素數量

前一個單詞詞性、音節數量

當前單詞詞性、音節數量、在短語中的位置、單詞數量、距離...等

下一個單詞詞性、音節數量

前一個短語中的音節數量、單詞數量

當前短語中的音節數量、單詞數量、在語句中的位置

下一個短語中的音節數量、單詞數量

此話語中的音節、單詞、短語數量

句子

Author of the danger trail

單詞

Author of the danger trail

音素

ao th er ah v

狀態

state1 state2 state3 state4 state5 state6 ... state20

幀



HMM 以5個狀態對齊示意圖

HMM 定義與三問題

HMM模型做了兩個很重要的假設如下(定義)

- 1) 任意時刻的隱藏狀態只依賴於它的前一個隱藏狀態
- 2) 觀測獨立性假設。任意時刻的觀察狀態只依賴當前時刻的隱藏狀態

HMM 三個基本問題

- 1) 評估觀察序列 (ex : MFCC特徵) 機率

- 即給定模型 λ ，計算在模型 λ 下觀測序列 O 出現的概率 P ，使用前向後向演算法

- 2) 模型參數學習問題

- 即給定觀測序列 O ，估計模型 $\lambda=(A,B,\Pi)$ 的參數，使該模型下觀測序列的條件概率 $P(O|\lambda)$ 最大，使用EM演算法

- 3) 預測問題，也稱為解碼問題

- 即給定模型 $\lambda=(A,B,\Pi)$ 和觀測序列 O ，求給定觀測序列條件下，最可能出現的對應的狀態序列，使用viterbi演算法

一、字母：

字母 26 個
母音 (a e i o u)
輔音 剩下21個

音素	分类	数量	示例
		48 个	全部音标
元音（音素）	单元音	12 个	[i:]、[ɜ:]等
	双元音	8 个	[eɪ]、[aɪ]等
辅音（音素）	清辅音	12 个	[p]、[t]等
	浊辅音	16 个	[b]、[d]等

二、音素(48=20+28)

音素是從音質角度劃分的最小的語音單位，從發音特徵上可分為兩類，即**母音**（也叫母音）**音素**和**輔音**（也叫子音）**音素**
英語中共有**48**個音素，其中母音**20**個，輔音**28**個。
字母是組成單詞的最小單位；音素是指字母在單詞中的讀音
blackboard只有**b-l-a-ck-b-oar-d**七個音素
用音標表示它們即[b] [l] [æ] [k] [b] [ɔ:] [d]

三、音標：

音標是**記錄音素的符號**，是音素的標寫符號。**它的制定原則是**：一個音素只用一個音標表示，而**一個音標並不只表示一個音素**
（雙母音就是由2個音素組成的，相對於單母音來說。由2個音素構成的音標我們稱之為雙母音）
注意：音標≠音素，音素是音，音標是符號。

四、音節

母音音素特別響亮，一個母音音素可構成一個音節，一個母音音素和一個或幾個輔音音素結合也可以構成一個音節。一般說來，母音音素可以單獨構成音節。
輔音音素不響亮，一般不能單獨構成音節。但英語輔音音素中有 4 個輔音[m]，[n]，[ŋ]，[l]是響音，它們和輔音音素結合，也可構成音節。它們構成的音節往往出現在詞尾，一般是非重讀音節。

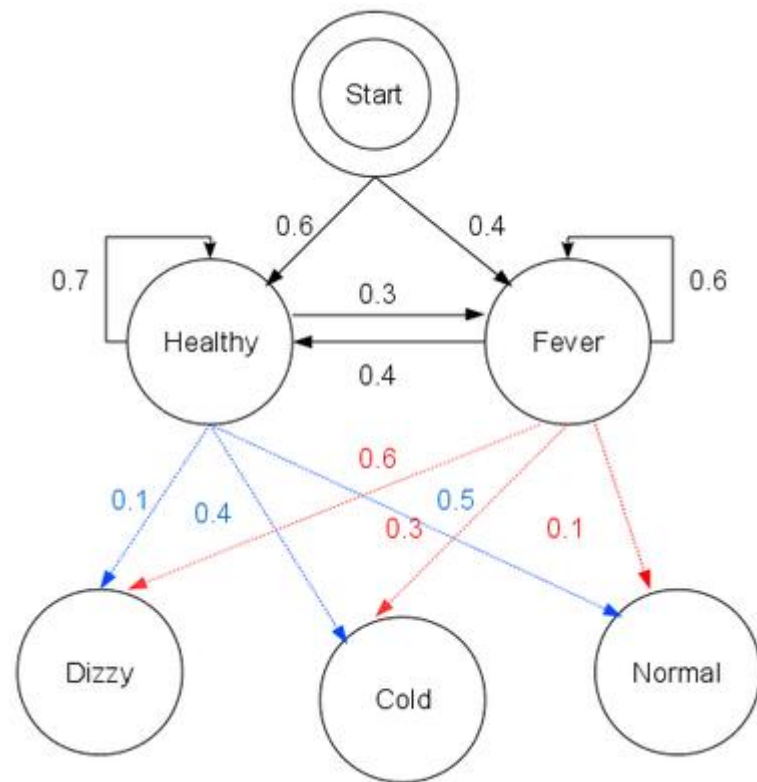
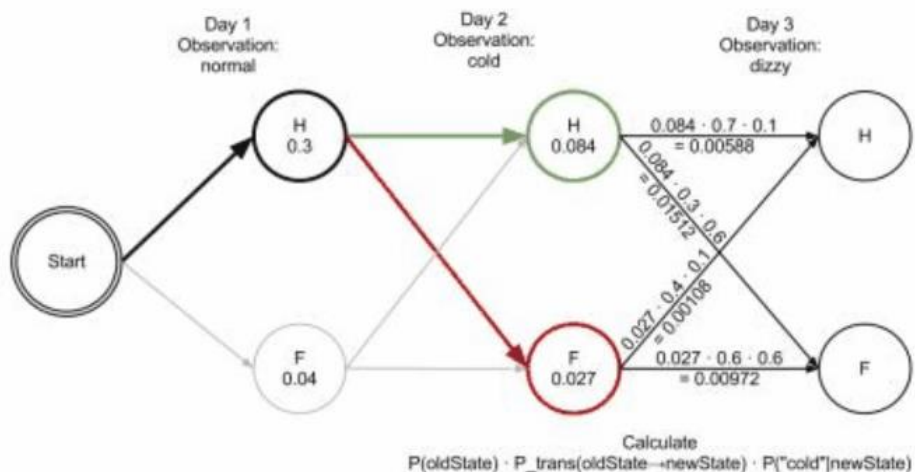
Viterbi演算法概念

Viterbi是一種動態規劃演算法。它用於尋找最有可能產生觀測事件序列的路徑，以及其機率。

今天有一位醫生要判斷病人是健康的還是發燒，病人只會回答正常、頭暈、冷，醫生要從這3個答案中判斷是否發燒，是否發燒就是隱藏狀態(無法直接觀察到)
右圖為病人各狀態的機率：

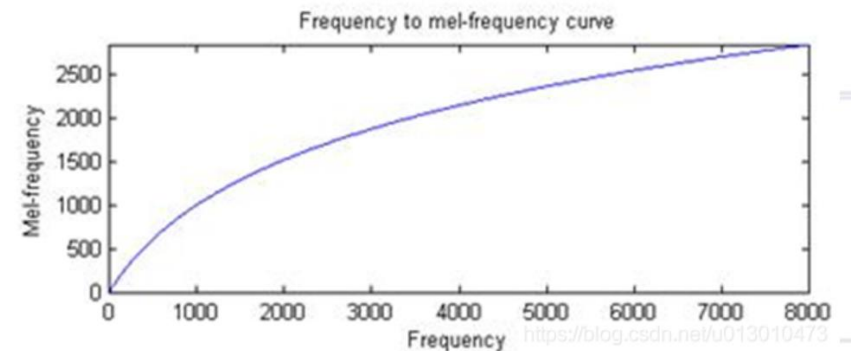
- 當天健康的病人隔天只會有30%的機率會發燒
- 如果病人是健康的會有50%的機率覺得正常
- 如果病人發燒了會有60%的機率覺得頭暈

病人連續看醫生3天，得以下結果：[正常、冷、頭暈]
根據viterbi演算法可以計算出3天的狀態分別是：[健康、健康、發燒]



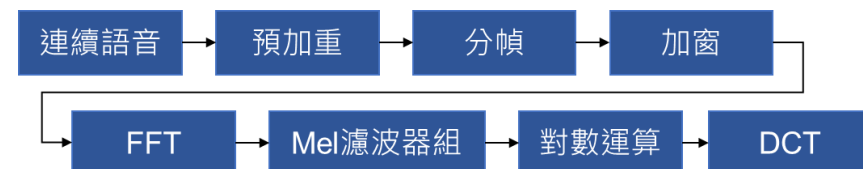
■ 提取MFCC

梅爾到頻譜係數(MFCC) 是在Mel刻度頻率域提取出來的倒譜參數，
Mel描述了人耳頻率的非線性特性
右圖展示Mel頻率與線性頻率的關係



- 預加重：通過一個高通濾波器，用來補償高頻
- **梅爾刻度濾波器過濾**，將訊號進行一個平滑，分成幾個子帶。
一般有兩種
 1. 三角帶通濾波器
 2. 高度的梅爾濾波
- **對數能量**：計算每個濾波器組輸出的對數能量，即子帶能量

MFCC的物理含義就是將語音物理資訊（頻譜包絡和細節）進行編碼運算得到的一組特徵向量，表示訊號頻譜的能量在不同頻率區間的分佈。



■ 提取F0

提取一幀聲音基頻的方法，大致可以分為**時域法**和**頻域法**。

- **時域法**以聲音的波形為輸入，其基本原理是**尋找波形的最小正週期**。當然，實際訊號的週期性只能是近似的。
- **頻域法**則會先對訊號做**傅裡葉變換**，得到頻譜（僅取**幅度譜**，捨棄相位譜）。頻譜上在基頻的整數倍處會有尖峰，頻域法的基本原理就是要求出這些尖峰頻率的最大公約數

提取F0的演算法：

DIO (時域法)

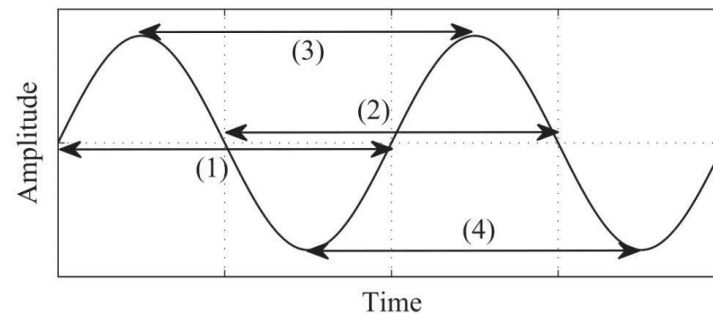
通過低通濾波器對原始訊號進行基頻的提取，具體流程是取4個週期計算標準差，並選最低的作為基頻

YIN (時域法)

時域法的基本原理是尋找波形的最小正週期。換句話說，就是看訊號平移多少後，與原訊號的重合度最高。

「重合度」有兩種定量衡量的方法，可以使用乘法的方式，亦可用減法的方式

YIN 演算法的名稱取自「陰陽」之「陰」，它表明演算法的核心思想是在差函數上尋找「穀值」，而不是在自相關函數上尋找「峰值」。



■ 動態特徵

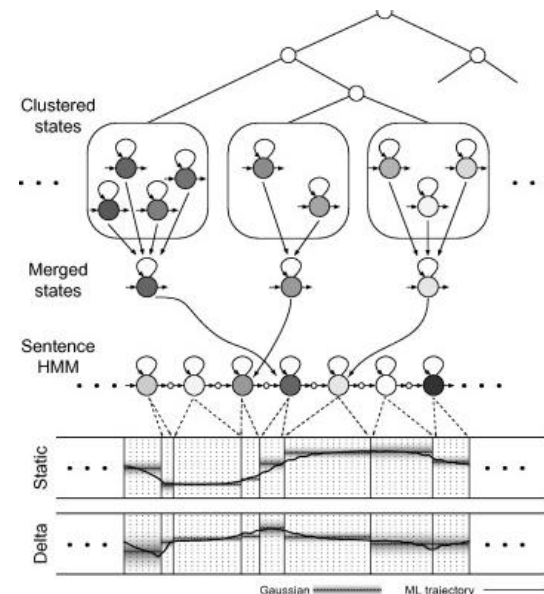
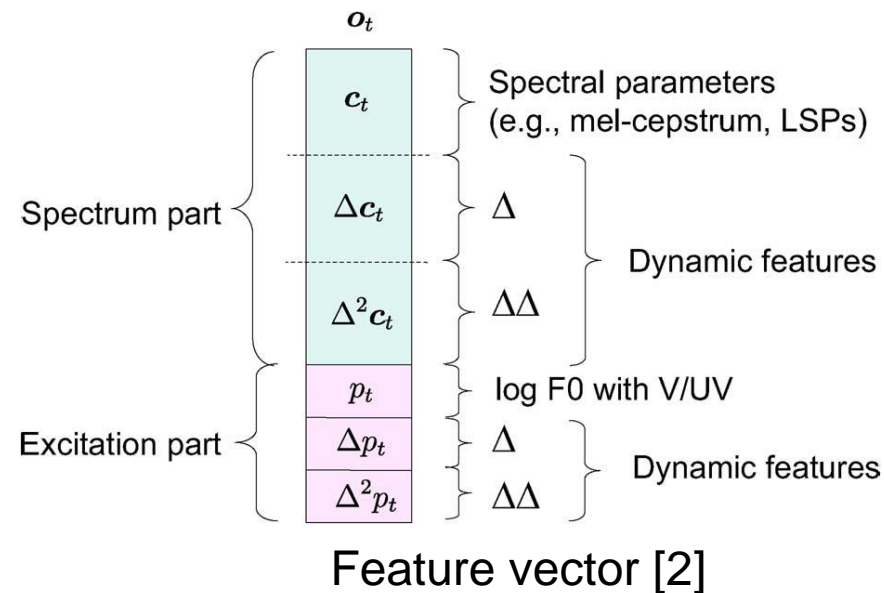
動態特徵為聲學特徵在相鄰幀間的變化情況。

透過HMM生成每個狀態的最有可能的觀測值，它只有考慮統計參數最有可能的觀測值，也就是該狀態的高斯分佈均值，但這種方式生成出來的參數會有明顯的分段，因此合成出來的音檔會不自然，因此引入動態特徵後可以有效改善此問題，讓合成出來的音檔平滑化。

O_t ：狀態輸出向量
 C_t ：M維靜態特徵
 ΔC_t ：一街動態特徵
 W ：是一個將動態特徵附加到
 c 的(單位、零)矩陣

$$\begin{bmatrix} o \\ \vdots \\ c_{t-1} \\ \Delta c_{t-1} \\ c_t \\ \Delta c_t \\ c_{t+1} \\ \Delta c_{t+1} \\ \vdots \end{bmatrix} = \begin{bmatrix} \dots & \vdots & \vdots & \vdots & \vdots & \dots \\ \dots & 0 & I & 0 & 0 & \dots \\ \dots & -I & I & 0 & 0 & \dots \\ \dots & 0 & 0 & I & 0 & \dots \\ \dots & 0 & -I & I & 0 & \dots \\ \dots & 0 & 0 & 0 & I & \dots \\ \dots & 0 & 0 & -I & I & \dots \\ \dots & \vdots & \vdots & \vdots & \vdots & \dots \end{bmatrix} \begin{bmatrix} c \\ \vdots \\ c_{t-2} \\ c_{t-1} \\ c_t \\ c_{t+1} \\ \vdots \end{bmatrix}$$

O_t 為 C_t 的線性變換



動態特徵的影響，使軌跡變平滑

■ 似然函數

似然估計提供了一種給定觀察資料來評估模型參數的方法

若有一函數： $P(x|\theta)$ 輸入有兩個： x 表示某一個具體的資料； θ 表示模型的參數

- 如果 θ 是已知確定的， x 是變數，這個函數叫做**概率函數**(probability function)，它描述對於不同的樣本點 x ，其出現概率是多少。
- 如果 x 是已知確定的， θ 是變數，這個函數叫做**似然函數**(likelihood function)，它描述對於不同的模型參數，出現 x 這個樣本點的概率是多少。

假設一三音素模型基於多元單高斯分佈，則可以計算出它的平均值和變異數，接著就可以套用右圖的公式計算出該模型在高斯上的似然值。

■ Log-likelihood for diagonal covariance Gaussian:

$$\begin{aligned} L(S) &= -\frac{1}{2} \sum_{i=1}^m \left[\sum_{k=1}^N \log(2\pi\sigma_k^2) + \sum_{k=1}^N \frac{(x_{ik} - \mu_k)^2}{\sigma_k^2} \right] \\ &= -\frac{1}{2} \left[m \sum_{k=1}^N \log(2\pi\sigma_k^2) + m \sum_{k=1}^N \frac{\sigma_k^2}{\sigma_k^2} \right] \\ &= -\frac{1}{2} \left[mN(1 + \log(2\pi)) + m \sum_{k=1}^N \log(\sigma_k^2) \right]. \end{aligned}$$

對所有資料計算似然值，並取 log

最佳的分裂問題

- 假設使用某一個問題可以將某一節點所對應的特徵分為兩部分 (l 和 r)，則根據分裂前和分裂後的似然變化去做判斷，下圖第一個公式為分裂後的似然和 $L(S_l)+L(S_r)$ ，另外扣掉分裂前似然 $L(S)$ ，如果算出來的值越大，則代表兩數據之間差距越大，則代表這個問題越好。
- 最佳分裂問題(公式二)為分裂前和分裂後的似然變化的簡化， $L(S_l)+L(S_r)-L(S)$

■ Log-likelihood difference:

$$L(S_l) + L(S_r) = -\frac{1}{2}mN(1 + \log(2\pi)) - \frac{1}{2}\left[m_l \sum_{k=1}^N \log(\sigma_{lk}^2) + m_r \sum_{k=1}^N \log(\sigma_{rk}^2)\right].$$

■ Best question:

$$q^* = \operatorname{argmin}_q \left[m_l \sum_{k=1}^N \log(\sigma_{lk}^2) + m_r \sum_{k=1}^N \log(\sigma_{rk}^2) \right],$$

$$\text{with } \sigma_{lk}^2 = \frac{1}{m_l} \sum_{x \in S_l} x_k^2 - \frac{1}{m_l^2} \left(\sum_{x \in S_l} x_k \right)^2$$

$$\sigma_{rk}^2 = \frac{1}{m_r} \sum_{x \in S_r} x_k^2 - \frac{1}{m_r^2} \left(\sum_{x \in S_r} x_k \right)^2.$$

■ 參數生成演算法

目標是在給定高斯分佈序列的前提下，計算出具有最大似然函數的語音參數序列。
最大似然估計方法是在近似無限多種模型參數中，想辦法導出最有可能產生與真實觀察到的資訊一樣的結果模型參數。

計算高斯分佈 (常態分佈) 需要有平均數與標準差。

目標找到一個 μ (平均數) 與 σ (標準差)，使目標樣本資訊代入時，使下方的概似函數值最大化

$$\begin{aligned}f(x | \mu) &= \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \\f(x_1, x_2 | \mu) &= \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_1-\mu)^2}{2\sigma^2}} * \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_2-\mu)^2}{2\sigma^2}} \\f(x_1, x_2, \dots, x_n | \mu) &= \prod_{i=1}^n \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}\end{aligned}$$

概似函數