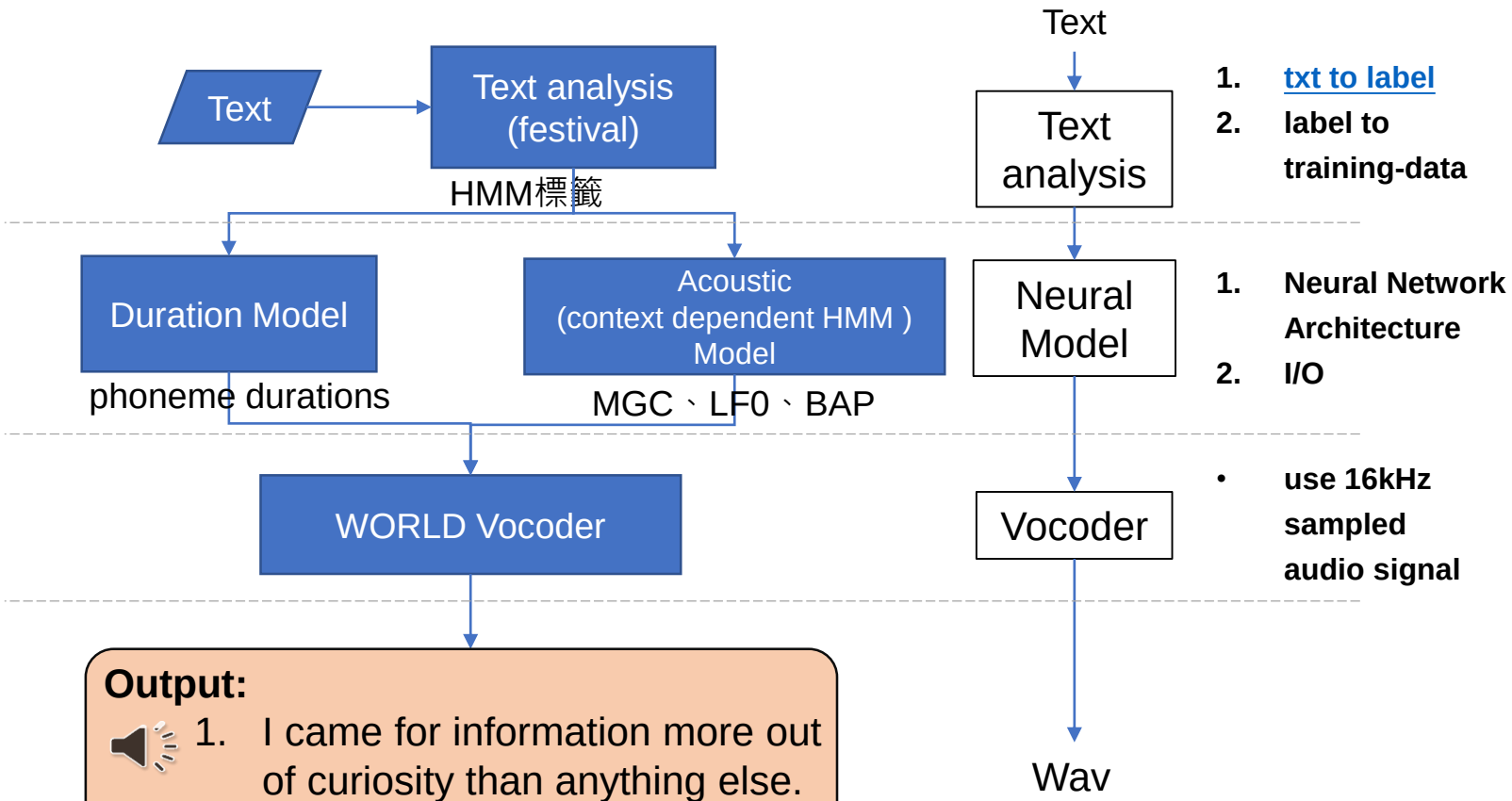
Traditional TTS design (block diagram with clear I/O & processing of each block)

Sian-Yi Chen

Advisors : Tay-Jyi Lin and Chingwei Yeh

Traditional TTS design

- block diagram with clear I/O & processing of each block



Output:

1. I came for information more out of curiosity than anything else.
2. Anyway, no one saw her like that.

- Advantages and disadvantages

統計參數合成 (Statistical Parametric Synthesis)

- 優：與更早以前的方法拼接合成(Unit Selection Synthesis) 比較，生成音檔更自然、更靈活方便修改參數、比串接式合成成本更低，不須要大量資料庫

- **缺：**
  1. **生成的語音還是具有較低的理解性**
  2. **很容易與人聲作區別**
  3. **像機器人的聲音**
  4. **在文本處理階段需要具備語言學、聲學的先備知識**

Text analysis (txt to label)

~~在Merlin提供的標籤中有兩種類別，分別是state align用狀態對齊與phoneme align用音素對齊，預設使用state align方式對齊。~~

~~State align使用HTK(Speech Recognition Toolkit)生成，以發音狀態為單位的標籤文件，而每個音素都由多個狀態組成，這邊則是指定生成5個HMM狀態的標籤。~~

txt to label

生成txt檔(一)，並使用festvox中一個名為EHMM的工具生成全文標籤，

EHMM(ergodic HMM)是一種對齊方法，它解釋了音素標籤之間可能存在停頓的可能性。

將txt檔轉換成全文標籤後，並依照HMM-base標籤格式(二)生成具有5個狀態得HMM標籤(三)

(一)：Text 檔

- (arctic_a0001 "Author of the danger trail, Philip Steels, etc.")
- (arctic_a0002 "Not at this particular case, Tom, apologized Whittemore.")

(二)：Context-dependent label format for HMM-based speech synthesis in English

p1^p2-p3+p4=p5@p6 p7
/A:a1 a2 a3
/B:b1-b2-b3@b4-b5&b6-b7#b8-b9$b10-b11!b12-b13;b14-b15|b16
/C:c1+c2+c3
/D:d1 d2 /E:e1+e2@e3+e4&e5+e6#e7+e8 /F: f1 f2
/G:g1 g2 /H:h1=h2^h3=h4|h5 /I:i1=i2
/J: j1+ j2- j3

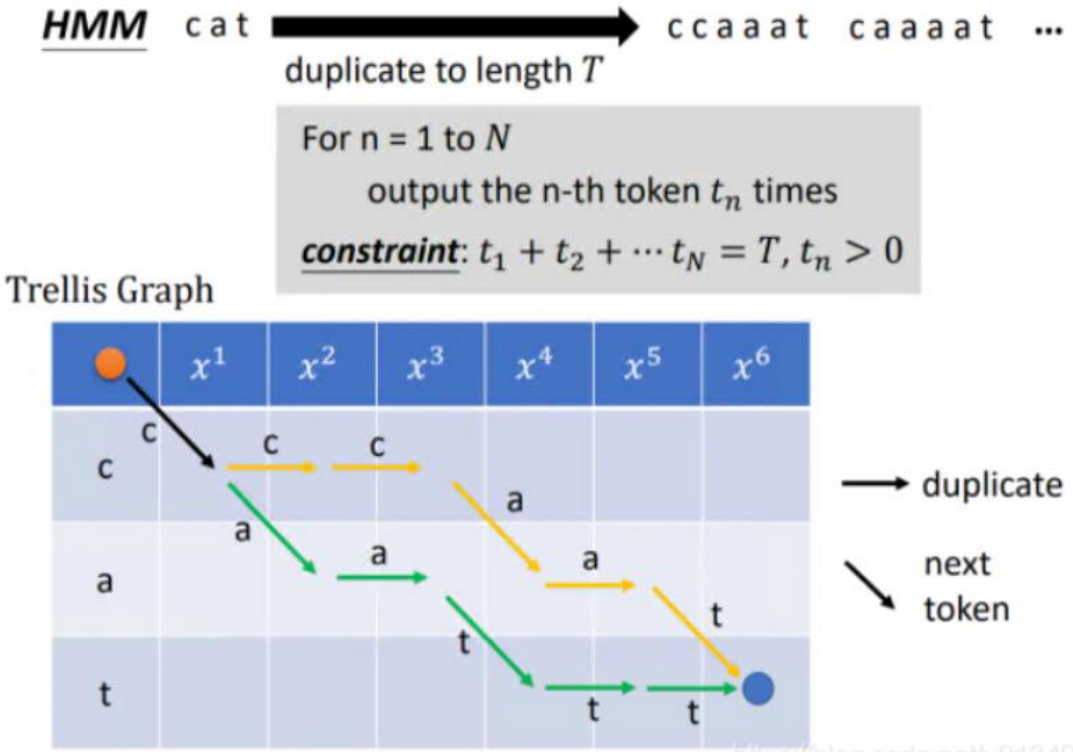
lab_format.pdf

(三)：具 5 個狀態的 HMM 標籤

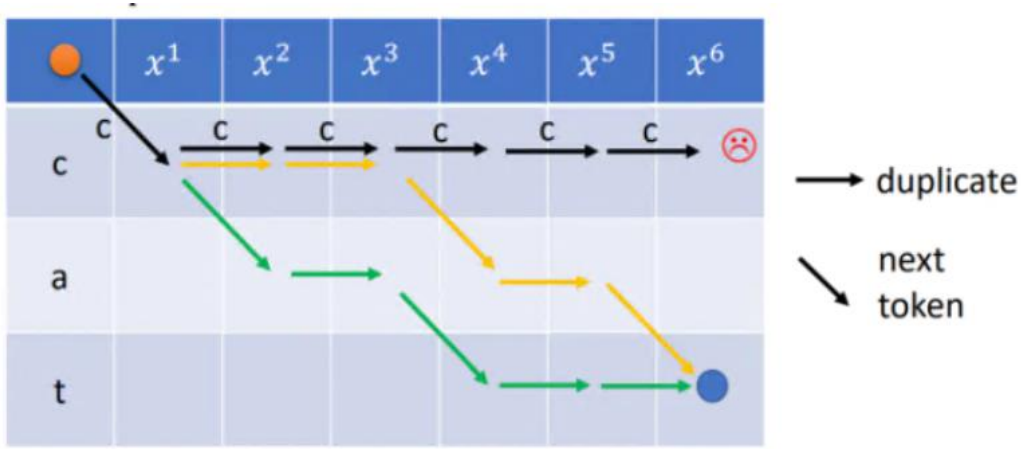
- 0 50000 x^x-sil+sil=ao@x_x/A:0_0_0/B:x-x-x@x-x&x-x#x-x$x-x!x-x;x-x|x/C:0+0+0/D:0_0/E:x+x@x+x&x+x#x+x/F:0_0/G:0_0/H:x=x@1=2|0/I:0=0/J:14+8-2[2]
- 50000 100000 x^x-sil+sil=ao@x_x/A:0_0_0/B:x-x-x@x-x&x-x#x-x$x-x!x-x;x-x|x/C:0+0+0/D:0_0/E:x+x@x+x&x+x#x+x/F:0_0/G:0_0/H:x=x@1=2|0/I:0=0/J:14+8-2[3]

HMM alignment

HMM (Hidden Markov Model)



一個token可以重複N次，但是所有token重複的次數和要等於acoustic features的長度T，也就是灰色部分所描述的公式。
表中橫軸代表acoustic features，縱軸表示token。
從左上角開始走到右下角，每步只能有兩個方向：向右走或是向下走，但終點一定要走到右下角才算是合法的路徑。
從起點一直走到終點所有合法的路徑就是所有可能的alignment。



Text analysis (label to training-data)

進神經網路訓練之前，需要將標籤檔轉換成二進位檔或是向量化，也就是現在神經網路做的Embedding，在Merlin中有兩種轉換的文件，差別為生成檔案的維度不同，分別為416與600維，此文件稱為**問題集(Question file)**。
問題集針對不同的語言需要自行設計，這邊使用的是416維的問題集，也就是由416道題目所組成，內容包含判斷前後文的聲韻母為何？聲母、韻母、韻律、位置特徵劃分等等。

Neural Network Architecture

Input: 416 dimensions label binary file
Duration model (DNN): 4*512 (tanh)

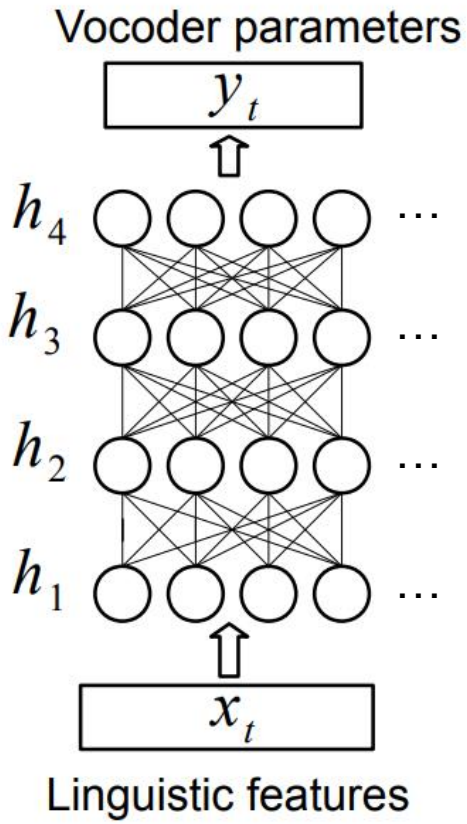
- **Output: 預測出每個音素 5 個狀態的持續時間**

- Batch size: 256
- Learning rate: 0.002
- Train file number: 50
- Valid file number: 5
- Test file number: 5

Acoustic model (DNN): 4*512 (tanh)

- **Output: mgc: 60維; bap: 1維; lf0: 1維;**

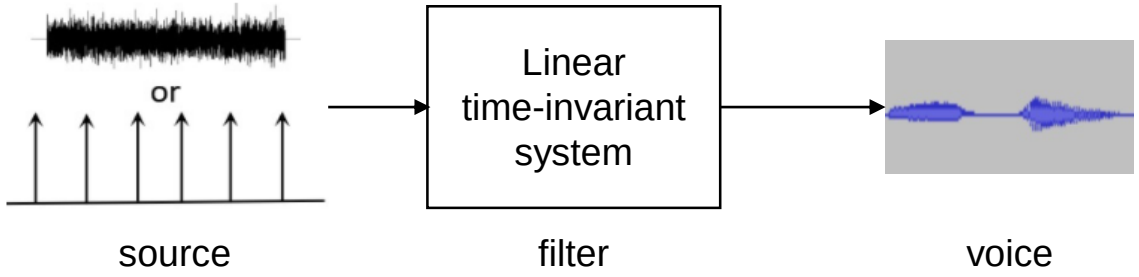
- Batch size: 64
- Learning rate : 0.002
- Train file number: 50
- Valid file number: 5
- Test file number: 5



圖一：前饋神經網路(DNN)

WORLD Vocoder

WORLD聲碼器是一個純訊號處理的聲碼器，可說是一個source-filter model

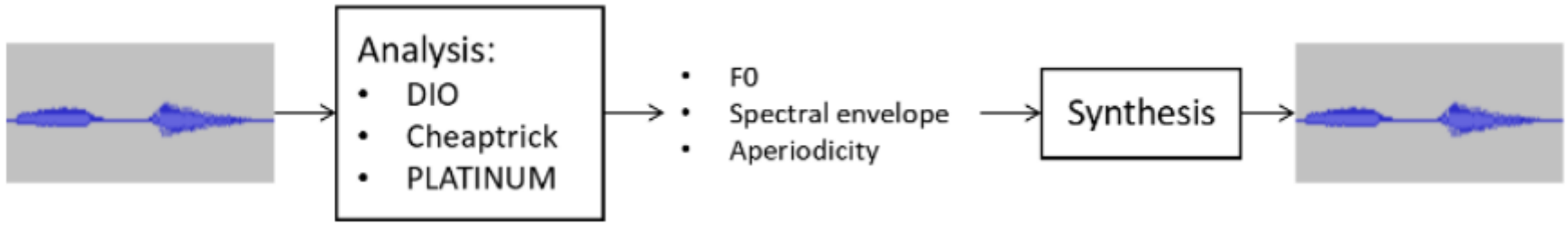


WORLD的輸入為每一個frame的基頻(F0)、頻譜包絡（spectral envelope）和非週期比值（aperiodicity）

- 基頻：決定了脈衝序列中的各個脈衝的位置
- 頻譜包絡(幅度譜的包絡曲線)：描述了該幀的音色和語音的內容，是線性時不變系統的頻率回應
- 非週期比值：描述了白色雜訊與脈衝序列能量的比例

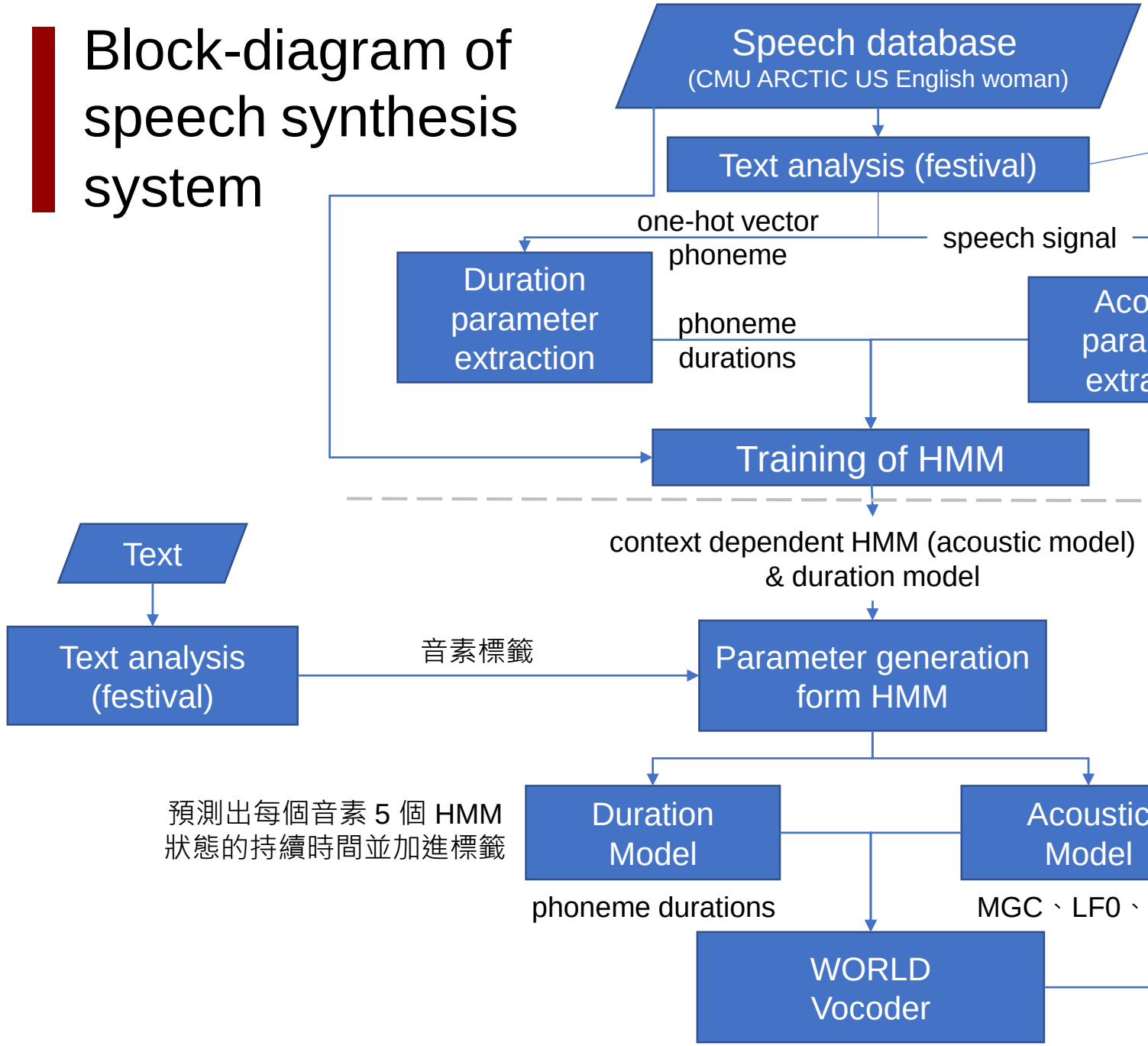
WORLD聲碼器的流程分3步：

1. 求各個脈衝的位置。
2. 對每個脈衝，求對應的脈衝回應。
3. 將所有脈衝回應疊加起來，得到語音波形。



附錄

Block-diagram of speech synthesis system



1. 生成HTS(HMM-based speech synthesis system)格式標籤
2. 生成每個音素 5 個 HMM 狀態的標籤
3. 將標籤對齊 phoneme
4. 使用問題集生成問題 (ex：聲母韻母？詞性？重音？等等)，同時設定神經網路維度

- MGC (Mel-generalized Cepstrum),
- BAP (band aperiodicity),
- VUV (voiced or unvoiced),
- LF0 (log f0)

Output:

1. I came for information more out of curiosity than anything else.
2. Anyway, no one saw her like that.

Innovations of Wang TTS (based on VCC 2020 ref. design)

Sian-Yi Chen

Advisors : Tay-Jyi Lin and Chingwei Yeh

Outline

- 目標：描述清楚傳統TTS的I/O，並且有上下關係的說明如何改善baseline的音質
  1. 傳統TTS
     - I/O與模塊處理
     - 傳統TTS架構與優缺
  2. 現今TTS
     - 傳統與現今TTS的區別
     - 現今架構的分類
     - 現今TTS的優缺點
  3. 基於VCC 2020 baseline TTS
     - 架構介紹
  4. VCC 2020 baseline TTS問題與改善
     - 問題說明
     - 改善方案
  5. 三點貢獻發想

Based on VCC 2020 ref. design

使用 VCC 2020 reference 的 VC baseline，架構由 2 個 model 組成，分別是 ASR + TTS，而兩個 model 互相獨立，ASR 將文字辨識出來後就可以與整個系統切開來，因此我們著重在 TTS model 的部分 (如圖一)。

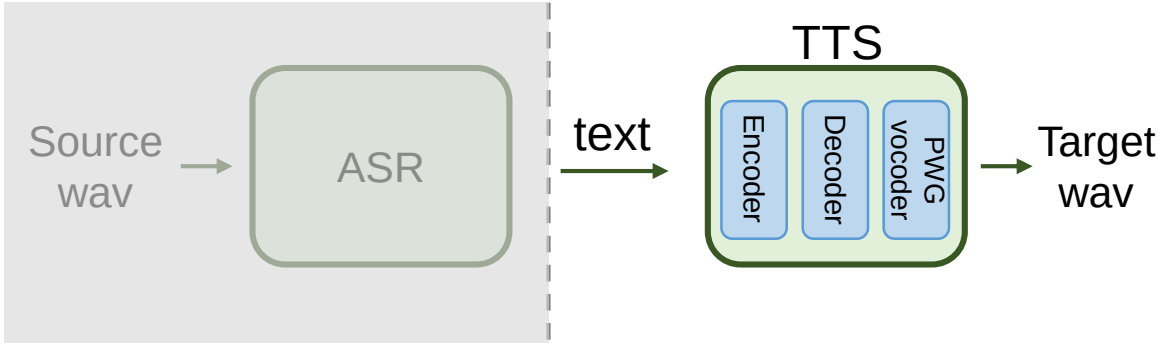
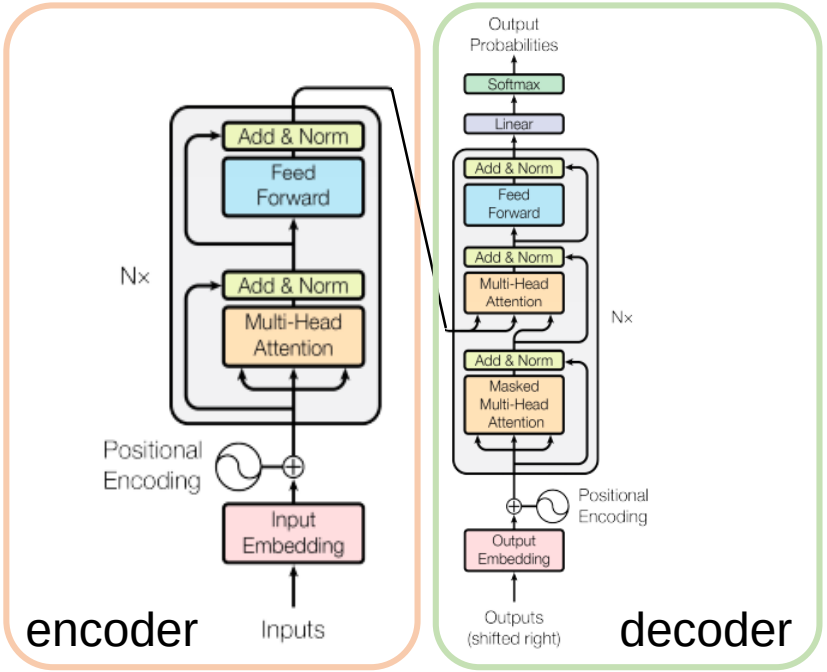


Figure 1: System structure

VCC2020 Baseline pre-training model: multi-speaker, x-vector Transformer-TTS model



架構可以拆解成以下四個區塊

1. Feature representation (MFCC)
   - 使用MFCC特徵作為輸入。
2. Embedding (x-vector)
   - 利用x-vector將輸入轉換成固定長度的特徵表示，也就是embedding的部分。
3. TTS model (Transformer)
4. Vocoder (Parallel WaveGAN)

VCC 2020 baseline TTS問題與改善

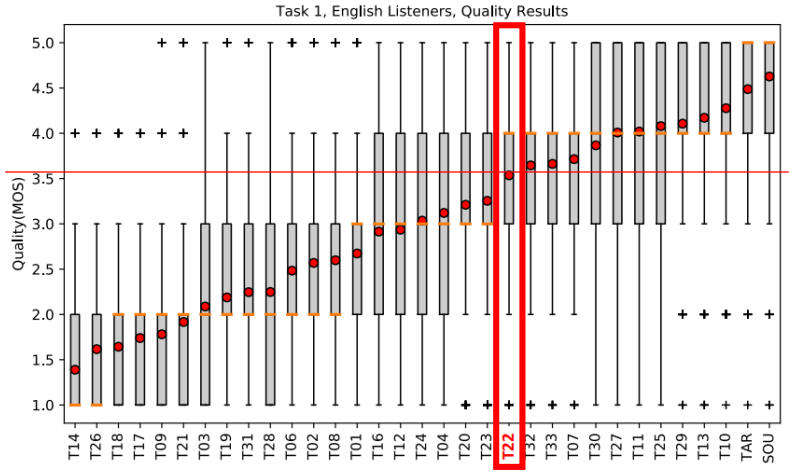
- 在論文中表示「ASR and TTS models still much room for improvement」，系統在VCC2020競賽中獲得3.5分的MOS分數，並獲得約90%的相似度

- 架構拆解成四個區塊
  1. Feature representation (MFCC)
  2. Embedding (x-vector)
  3. TTS model (Transformer)
  4. Vocoder (Parallel WaveGAN)

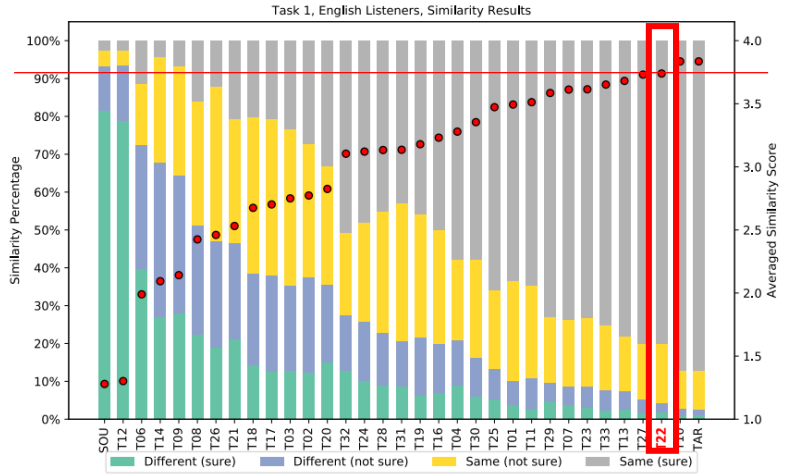
- 方案 (可以有效達到音效的改善)
  1. 認為將3或4使用的模型更改
  2. 提升目前使用的取樣率(16kHz)

- 原因
  1. Transformer算是較單純的模型
  2. 目前使用的Vocoder為non-AR，換成autoregressive model是普遍認知較好的方法

- 三點貢獻發想
  1. 合成出王老師的聲音
  2. 與baseline相比，音質提升
  3. 再次證明ASR+TTS這種cascade的方式是有競爭性的



(a) *Naturalness results for task 1.*



(b) *Similarity results for task 1.*