Introduction to Speech Synthesis

CPSC 503 Pedagogical Project

Junze Wu, Dec 2020

Overview

Speech Synthesis

- Also called Text-to-Speech (TTS)
- Artificial production of human speech (acoustic waveform) from text input

Overview

Applications

- Accessibility aid for people with vocal disabilities
- ACAT (Assistive Context-Aware Toolkit) designed by Intel

https://www.wired.com/2015/01/intel-gave-stephen-hawking-voice/

Overview

Applications

- Virtual assistants
- Smart speakers
- Navigation systems in cars
- …

Apple
Siri

Microsoft
Cortana

Google
Google Home

amazon alexa

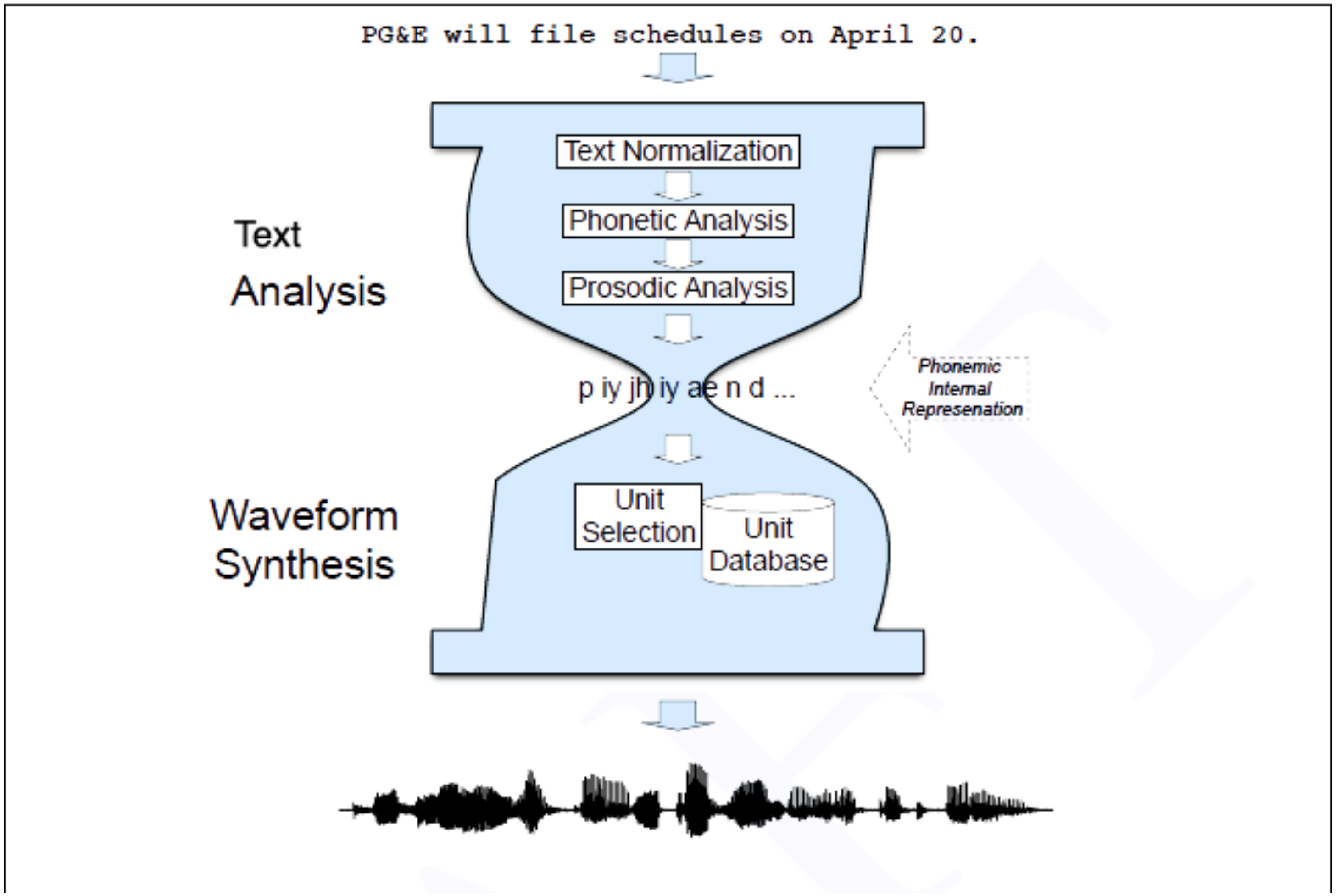
Outline
Two steps

- **Text analysis**: Convert text input into intermediate representation
- **Waveform synthesis**: Convert intermediate representation into waveform
- e.g. PG&E will file schedules on April 20.

* * * L-L%
P G AND E WILL FILE SCHEDULES ON APRIL TWENTIETH
p iy jh iy ae n d iy w ih l f ay l s k eh jh ax l z aa n ey p r ih l t w eh n t iy ax th

Outline

Two steps



Outline

Text Analysis

- Text Normalization
- Phonetic Analysis
- Prosodic Analysis

Outline

Waveform synthesis

- Concatenative synthesis
    - Select units from database of recorded speech and concatenate them together to generate speech
- Statistical parametric synthesis
    - Based on HMM
- End-to-end synthesis based on deep learning
    - WaveNet, Tacotron

Part I: Text Analysis

1. Text Normalization
2. Phonetic Analysis
3. Prosodic Analysis

Text Normalization

- Sentence Tokenization
- Non-Standard Words
- Homograph Disambiguation

Text Normalization

Sentence tokenization

- Determine boundary of sentences
- Disambiguation of period “.”
    - e.g. The group included Dr. J. M. Freeman and T. Boone Pickens Jr.
- Train a binary classifier using supervised machine learning (logistic regression/ SVM/decision tree, etc.)
- End-of-sentence (EOS) vs. not-EOS

Text Normalization
Sentence tokenization

- Features that we can consider:
- the prefix (the portion of the candidate token preceding the candidate)
- the suffix (the portion of the candidate token following the candidate)
- whether the prefix or suffix is an abbreviation (from a list)
- the word preceding the candidate
- the word following the candidate
- whether the word preceding the candidate is an abbreviation
- whether the word following the candidate is an abbreviation

Text Normalization
Sentence tokenization

- Example:

ANLP Corp. chairman Dr. Smith resigned.

The features for the period "." in the word Copr. would be:

Prefix = Corp, Suffix = NULL, PreviousWord = ANLP, NextWord = chairman,

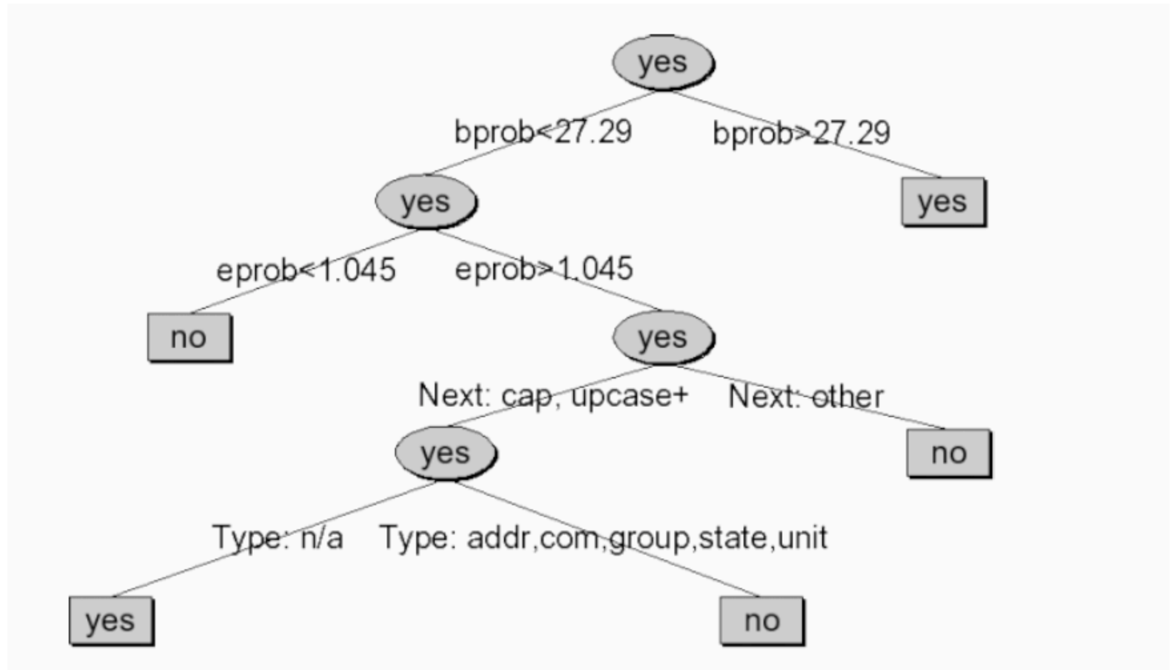
PreviousWordAbbreviation=1, NextWordAbbreviation=0

- Other features:
  - probability of occurring at beginning/end of sentence
  - case (sentence usually begin with capital letters)
  - Abbreviation subclass: titles (e.g. Mr., Dr., Gen.), months (e.g. Jan., Feb.)

Text Normalization

Sentence tokenization

- Decision tree
- yes = EOS, no = not-EOS
- Probability of beginning (bprob) or end (eprob) of sentence
- Case of next word (Next)
- Abbreviation class (Type)
- CART



Text Normalization

Non-Standard Words

ALPHA	EXPN	abbreviation	*adv, N.Y., mph, gov't*
	LSEQ	letter sequence	*DVD, D.C., PC, UN, IBM,*
	ASWD	read as word	*IKEA*, unknown words/names
NUMBERS	NUM	number (cardinal)	*12, 45, 1/2, 0.6*
	NORD	number (ordinal)	*May 7, 3rd, Bill Gates III*
	NTEL	telephone (or part of)	*212-555-4523*
	NDIG	number as digits	*Room 101*
	NIDE	identifier	*747, 386, I5, pc110, 3A*
	NADDR	number as street address	*747, 386, I5, pc110, 3A*
	NZIP	zip code or PO Box	91020
	NTIME	a (compound) time	*3.20, 11:45*
	NDATE	a (compound) date	*2/28/05, 28/02/05*
	NYER	year(s)	*1998, 80s, 1900s, 2008*
	MONEY	money (US or other)	*$3.45, HK$300, Y20,200, $200K*
	BMONEY	money tr/m/billions	*$3.45 billion*
	PRCT	percentage	*75% 3.4%*

Figure 8.4 from textbook SLP2

Text Normalization

Non-Standard Words

- Numbers (1750 => seventeen fifty/one seven five zero/seventeen hundred and fifty/one thousand seven hundred and fifty)
- Abbreviations (Jan 1 => January first)
- Letter Sequences (UN, DVD, PC, IBM)
- Acronyms (IKEA, NASA, UNICEF)

Text Normalization

Non-Standard Words

- Tokenization: identify NSWs in the input text
- Classification: classify NSWs into specific types
- Expansion: expand into ordinary words

Text Normalization

Non-Standard Words

- Splitter splits words into tokens by looking for whitespaces
- Some words needs to be cut into combinations (e.g. 2-car, RVing)
- Classification of NSW type can be done with regular expressions
  - e.g. NYER (years) /(1[89][0-9][0-9])|(20[0-9][0-9]/
- Or train a classifier
- Expansion is based on simple rules such as:
  - LSEQ expands to a sequence of words, one for each letter
  - ASWD expands to itself
  - NUM expands to a sequence of words representing the cardinal number
  - NYER expand to 2 pairs of NUM digits

Text Normalization
Homograph Disambiguation

Homographs: words with same spelling but different pronunciations

It’s no **use** (/y uw s/) to ask to **use** (/y uw z/) the telephone.

Do you **live** (/l ih v) near a zoo with **live** (/l ay v/) animals?

15 Most common homographs in order: use, increase, close, record, house, contract, lead, live, lives, protest, survey, project, separate, present, read

Text Normalization
Homograph Disambiguation

- Relationships between homographs:

Final voicing			Stress shift			-ate final vowel		
	N (/s/)	V (/z/)		N (init. stress)	V (fin. stress)		N/A (final /ax/)	V (final /ey/)
use	y uw s	y uw z	record	r eh1 k axr0 d	r ix0 k ao1 r d	estimate	eh s t ih m ax t	eh s t ih m ey t
close	k l ow s	k l ow z	insult	ih1 n s ax0 l t	ix0 n s ah1 l t	separate	s eh p ax r ax t	s eh p ax r ey t
house	h aw s	h aw z	object	aa1 b j eh0 k t	ax0 b j eh1 k t	moderate	m aa d ax r ax t	m aa d ax r ey t

Figure 8.5 from textbook SLP2

Text Normalization

Homograph Disambiguation

- Different forms of a homograph tend to have different part-of-speech
- use (noun: /y uw s/, verb: /y uw z/)
- live (noun: /l ay v/, verb: /l ih v/)
- Homograph Disambiguation can thus be solved as POS tagging problem
- Solved using a Hidden Markov Model and the Viterbi algorithm

Phonetic Analysis

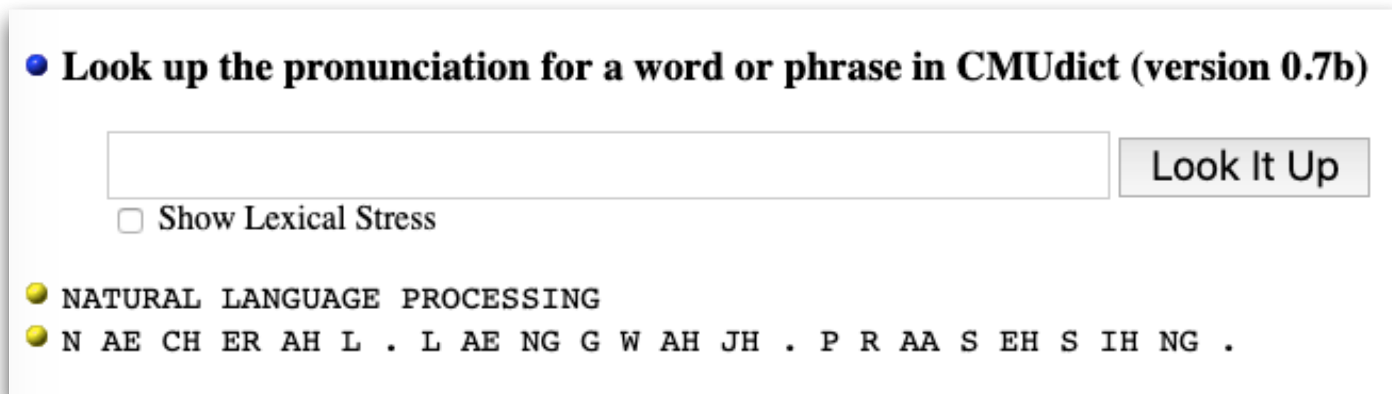
From words to phonemes

- Phoneme: a unit of sound
- Methods:
  - Look up a pronunciation dictionary
  - Grapheme-to-phoneme (g2p)

Phonetic Analysis

Phonemic Internal Representation

- ARPABET: an English phonetic alphabet
- CMU Pronunciation Dictionary
- http://www.speech.cs.cmu.edu/cgi-bin/cmudict?in=C+M+U+Dictionary



Phonetic Analysis
Phonemic Internal Representation

- Python package: cmudict
- cmudict.dict()
- cmudict.phones()

Phonetic Analysis

Phonemic Internal Representation

- 39 phonemes in total
- vowels carry a lexical stress marker: 0=no stress, 1=primary stress, 2=secondary stress
- Sample pronunciations from the CMU dictionary

ANTECEDENTS	AE2 N T IH0 S IY1 D AH0 N T S	*PAKISTANI*	P AE2 K IH0 S T AE1 N IY0
CHANG	CH AE1 NG	*TABLE*	T EY1 B AH0 L
DICTIONARY	D IH1 K SH AH0 N EH2 R IY0	*TROTSKY*	T R AA1 T S K IY2
DINNER	D IH1 N ER0	*WALTER*	W AO1 L T ER0
LUNCH	L AH1 N CH	*WALTZING*	W AO1 L T S IH0 NG
MCFARLAND	M AH0 K F AA1 R L AH0 N D	*WALTZING(2)*	W AO1 L S IH0 NG

Figure 8.6 from textbook SLP2

Phonetic Analysis

Grapheme-to-phoneme (g2p)

- Converting a sequence of letters into a sequence of phonemes
- Guess the pronunciation of words based on their spellings
- Early version: Handwritten letter-to-sound (LTS) rules
  - A[B]C=D means: B with left-context A and right-context C is pronounced as D
  - e.g. [CI]A=/SH/, as in asso**cia**tion
- Modern version: Train a decision tree or neural network to learn LTS rules
- Further reading: TTS textbook (chapter 8.4)

Prosodic Analysis

- Prosody: the study of intonational and rhythmic aspects of language
- Acoustic features: F0 (Fundamental frequency), intensity, duration
- Convey sentence-level pragmatic meanings

Prosodic Analysis

Three aspects of prosody

- Prominence: some words are more prominent than others
- Structure:
  - Some words tend to group together
  - Some words tend to have a noticeable break or disjuncture in between
- Tune: intonation
- Prosodic analysis: compute an abstract representation of prosodic prominence, structure, and tune

Prosodic Analysis
Structure

- Structure:
  - Some words tend to group together
  - Some words tend to have a noticeable break or disjuncture in between
- Prosodic phrasing: predict prosodic boundaries - a classification problem
- Implications:
  - The final vowel of a phrase is longer than usual
  - Insert pause after phrase
  - Often a slight drop in F0 from the beginning of phrase

Prosodic Analysis
Prominence

- Prominence: some words are more prominent than others
- Pitch accent: linguistic marker for prominent words

Prosodic Analysis

Tune

- Tune: the rise and fall of F0 over time
- Examples:
    - Question rise
    - Final fall

Prosodic Analysis

Tune

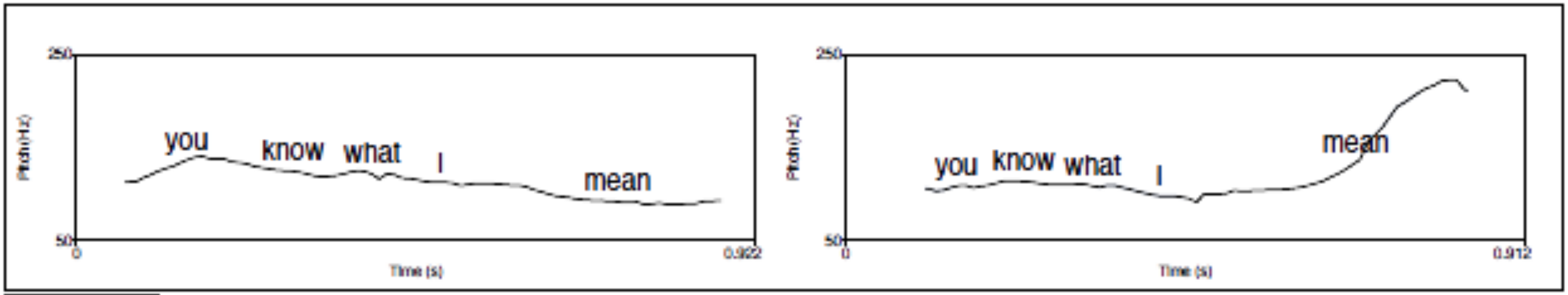


Figure 8.8 The same text read as the statement *You know what I mean.* (on the left) and as a question *You know what I mean?* (on the right). Notice that yes-no-question intonation in English has a sharp final rise in F0.

Prosodic Analysis

Tune

- More sophisticated intonation models
  - ToBI
  - Tilt

Pitch Accents		Boundary Tones	
H*	peak accent	**L-L%**	"final fall": "declarative contour" of American English"
L*	low accent	**L-H%**	continuation rise
L*+H	scooped accent	**H-H%**	"question rise": cantonical yes-no question contour
L+H*	rising peak accent	**H-L%**	final level plateau (plateau because H- causes "upstep" of following)
H+!H*	step down		

Recall: Intermediate Representation

Two steps

- **Text analysis**: Convert text input into intermediate representation
- **Waveform synthesis**: Convert intermediate representation into waveform
- e.g. PG&E will file schedules on April 20.

*   *   *   L-L%
P G AND E WILL FILE SCHEDULES ON APRIL TWENTIETH
p iy jh iy ae n d iy w ih l f ay l s k eh jh ax l z aa n ey p r ih l t w eh n t iy ax th

Part II. Waveform Synthesis

- Concatenative synthesis
  - Select units from database of recorded speech and concatenate them together to generate speech
- Statistical parametric synthesis
  - Based on HMM
- End-to-end synthesis based on deep learning
  - WaveNet, Tacotron

Concatenative Synthesis

- Diphone synthesis
- Unit selection synthesis

Concatenative Synthesis
Diphone Synthesis

- Diphone: a unit which starts in middle of one phone and extends to the middle of the next phone
- Diphone synthesis: generates waveform for a sequence by selecting from pre-recorded database of diphones
- To adjust for prosody, use TD-PSOLA:
  - Duration: duplicate/remove part of the signal
  - Pitch (F0): resample to change pitch

Concatenative Synthesis

Diphone Synthesis

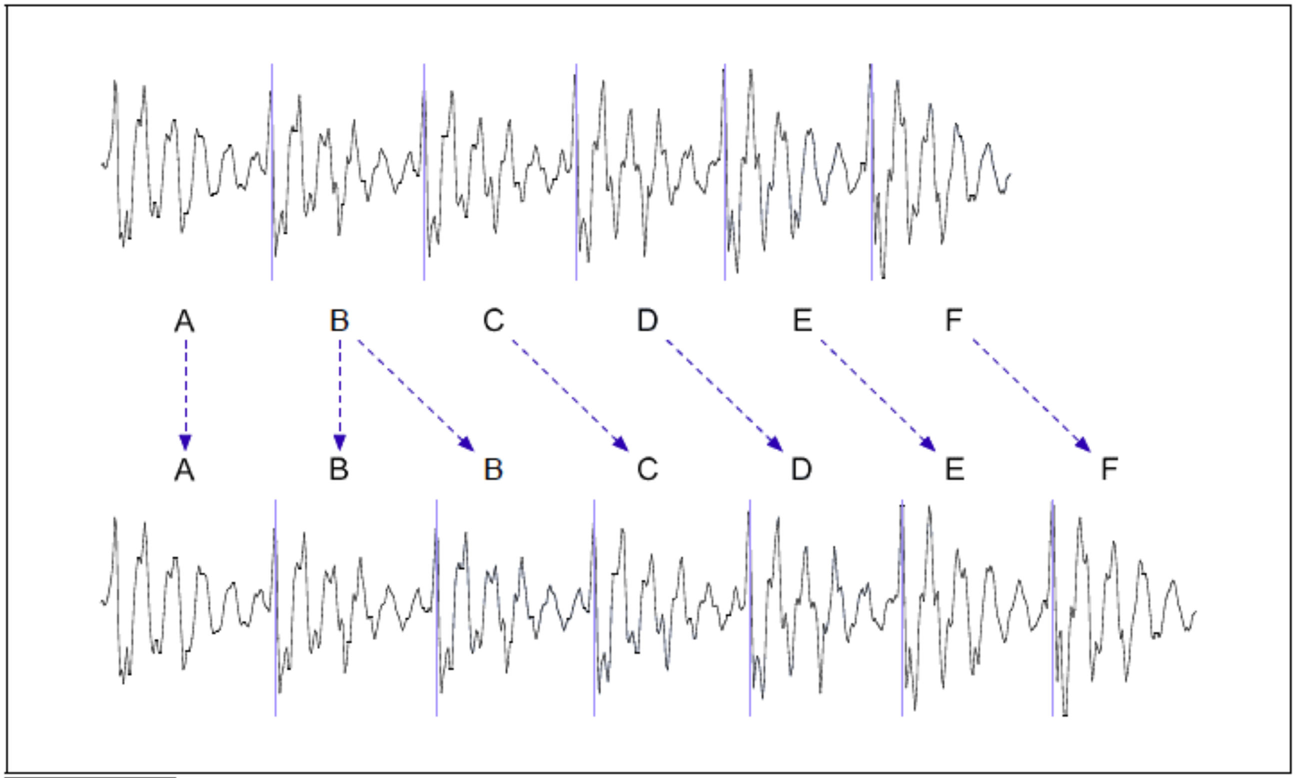


Figure 8.16 TD-PSOLA for duration modification. Individual pitch-synchronous frames can be duplicated to lengthen the signal (as shown here), or deleted to shorten the signal.

Concatenative Synthesis

Diphone Synthesis

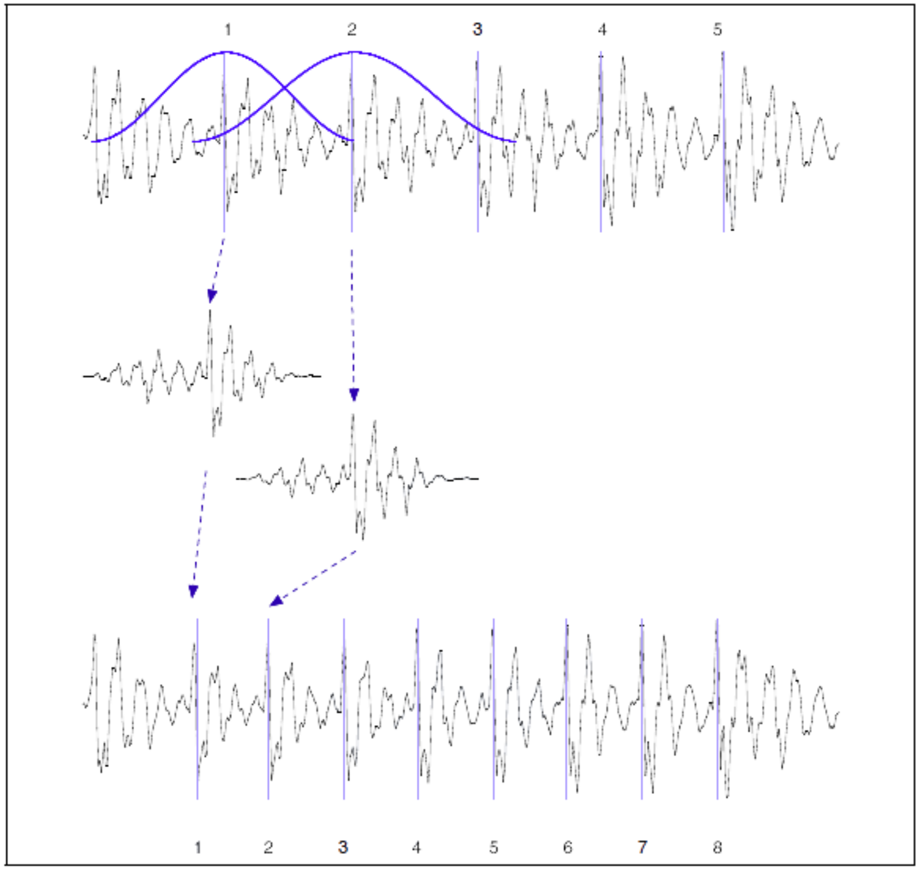


Figure 8.17 TD-PSOLA for pitch (F0) modification. In order to increase the pitch, the individual pitch-synchronous frames are extracted, Hanning windowed, moved closer together and then added up. To decrease the pitch, we move the frames further apart. Increasing the pitch will result in a shorter signal (since the frames are closer together), so we also need to duplicate frames if we want to change the pitch while holding the duration constant.

Concatenative Synthesis

Unit Selection Synthesis

- Use a much larger database that contains many copies of each diphone
- No signal processing required
- Select the “best” unit minimizing the sum of **target cost** and **join cost**
- Using a Hidden Markov Model:
    - target units are the observed outputs
    - the units in the database are the hidden states
- Solve for best path of hidden units using the Viterbi algorithm

Concatenative Synthesis

Unit Selection Synthesis

Target cost $T(u_t, s_t)$:	how well the target specification s_t matches the potential unit u_t
Join cost $J(u_t, u_{t+1})$:	how well (perceptually) the potential unit u_t joins with its potential neighbor u_{t+1}

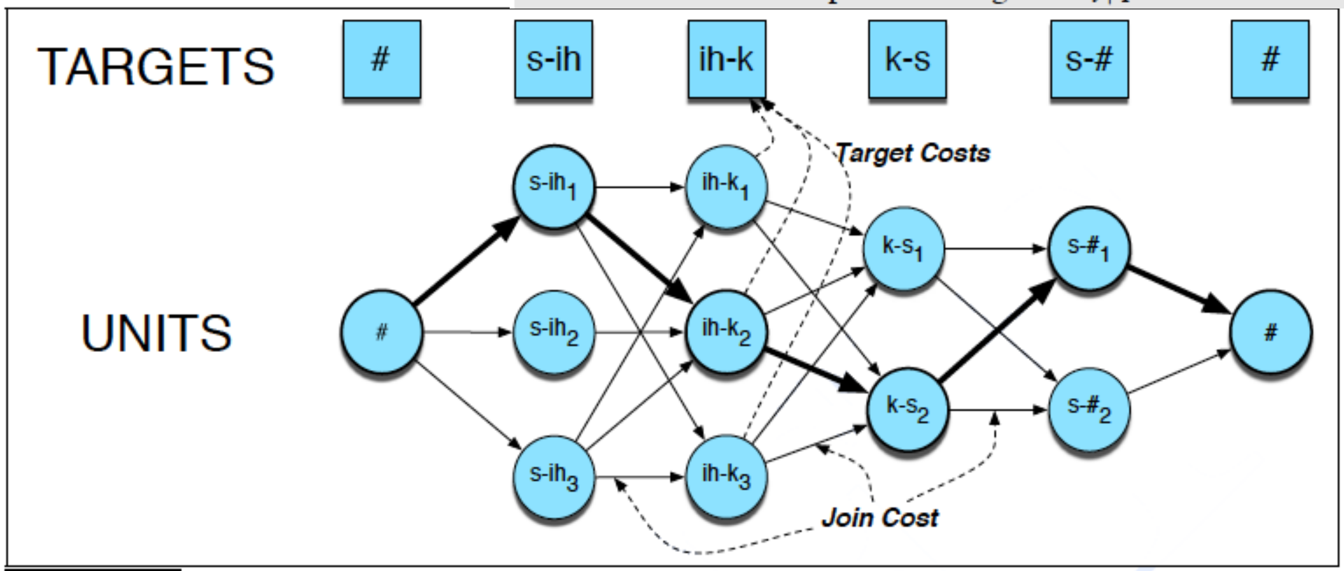


Figure 8.18 The process of decoding in unit selection. The figure shows the sequence of target (specification) diphones for the word *six*, and the set of possible database diphone units that we must search through. The best (Viterbi) path that minimizes the sum of the target and join costs is shown in bold.

Statistical Parametric Synthesis

- Learn instead of memorize
- Predict acoustic features from linguistic features
- Based on HMM
- Use a vocoder (voice encoder) to generate waveforms from acoustic features
- Require far less memory to store parameters than to memorize entire dataset

Statistical Parametric Synthesis

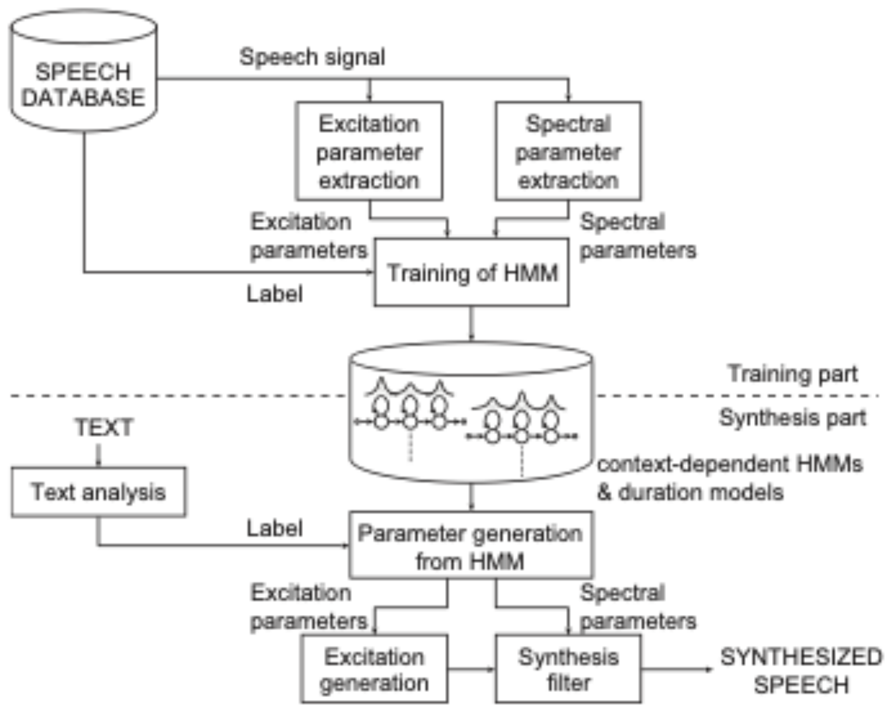


Fig. 1. Overview of a typical HMM-based speech synthesis system.

End-to-End Synthesis

- Learn a model directly from text input to wave features
- Based on deep learning
  - WaveNet
  - Tacotron

End-to-End Synthesis

WaveNet

- Based on dilated causal CNN (CNN with holes)
-

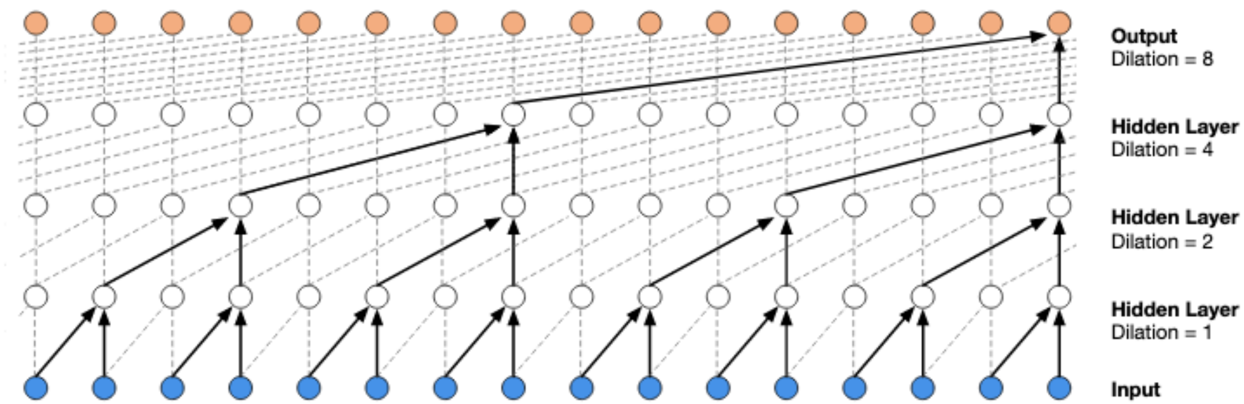


Figure 3: Visualization of a stack of *dilated* causal convolutional layers.

End-to-End Synthesis
Tacotron

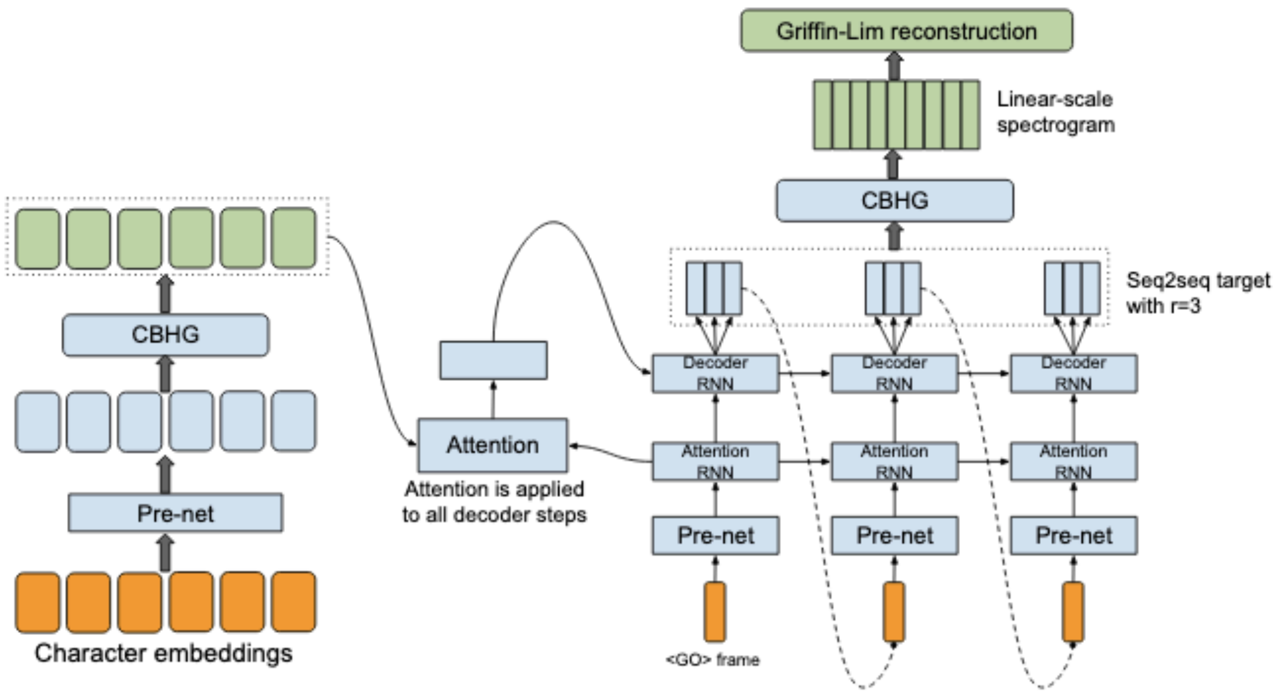


Figure 1: *Model architecture. The model takes characters as input and outputs the corresponding raw spectrogram, which is then fed to the Griffin-Lim reconstruction algorithm to synthesize speech.*

End-to-End Synthesis

Tacotron

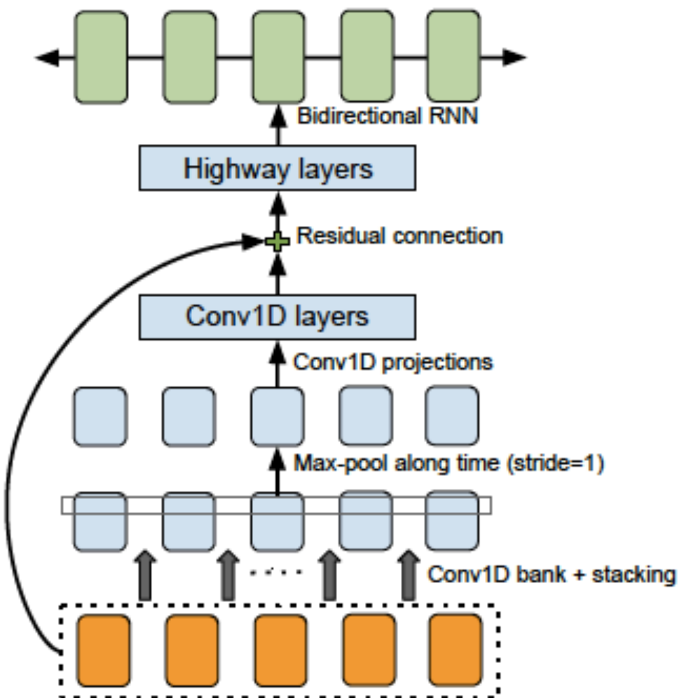


Figure 2: The CBHG (1-D convolution bank + highway network + bidirectional GRU) module adapted from Lee et al. (2016).

Summary
Speech Synthesis

- Text analysis
  - Text normalization
  - Phonetic analysis
  - Prosodic analysis
- Waveform synthesis
  - Concatenative
  - Parametric
  - End-to-end

References and Readings

- *Speech and Language Processing An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition* by Daniel Jurafsky, James H. Martin (Textbook, 2nd edition, Chapter 8)
- *Text-to-speech Synthesis* by Paul Taylor
- Statistical parametric speech synthesis (Review paper) https://www.sciencedirect.com/science/article/abs/pii/S0167639309000648
- *Speech Synthesis Based on Hidden Markov Models* (Research paper) https://ieeexplore.ieee.org/document/6495700?arnumber=6495700
- WaveNet: A generative model for raw audio (Blog post) https://deepmind.com/blog/article/wavenet-generative-model-raw-audio
- Tacotron: Towards End-to-End Speech Synthesis (Research paper) https://google.github.io/tacotron/
- Slides from course LSA 352 https://nlp.stanford.edu/courses/lsa352/

Learning Goals

1. Be able to identify the problems that need to be addressed during the text analysis step
2. Understand the internal phonetic representation and ARPABET
3. Explain the differences between concatenative, parametric, and end-to-end synthesis
4. Be able to perform basic signal processing to synthesize waveforms using Python

Assignment

1. Come up with example sentences that might cause issues during text analysis and then try the examples using a widely available TTS service such as IBM's Watson. Report any errors and explain how the issues can be resolved.

2. Practice using the CMU Pronouncing Dictionary to convert between the internal ARPABET representation of words and plain text (e.g. COMPUTER and K AH M P Y UW T ER).

3. Assigned readings. Write a brief summary of different synthesis methods and explain their differences.

4. Coding assignment: implement a simple concatenative text-to-speech program using Python

Questions?

Links

- Lecture recording: https://youtu.be/sMWv5S678kA
- Assignment Code repo: https://github.com/junzew/cpsc503-project-assignment