Wordcount - 1998

Contents

1. Introduction

This report aims to provide account of speech synthesis process conducted by Festival (Black et al. 2014) by giving an overview of the components of a Text-To-Speech (tts) system, actual process followed through for festival and listing, explaining, and providing solutions to errors produced in the process.

2. Background

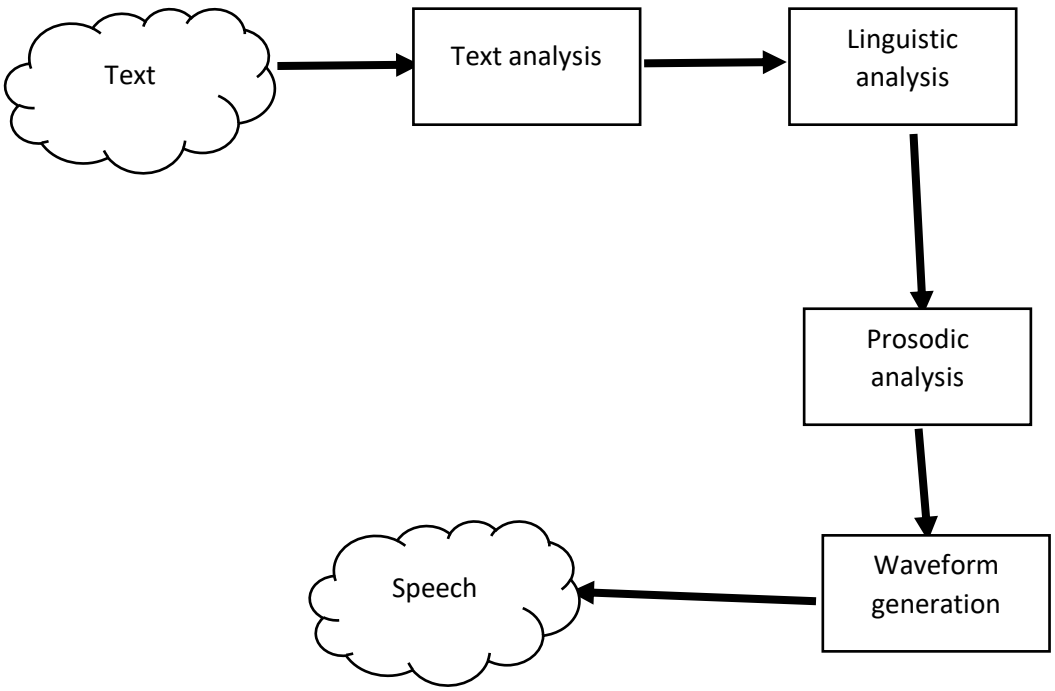

Figure 1: Simple outline on TTS pipeline.

2.1 Human speech production

Speech sounds are created by pushing air from lungs up and outward through vocal fold vibration in larynx, other sounds synthesised through moving trapped air within vocal tract via tongue. Fundamental frequency F0 is the rate at which vocal folds vibrate. Sounds are categorized into categories – vowels and consonants. Vowels are voiced sounds (vocal fold vibration) whereas consonants maybe voiced or voiceless. Phonemes are smallest unit of sounds in a language. Phone is the actual sound of a word you can hear. Diphones are adjacent pair of phones in an utterance. “TTS use diphones as it captures co-articulation (changes in speech articulation due to neighbouring speech)” (Hood, 2004).

2.2 Text processing

Text module handles whitespace tokenisation by separating text into independent utterances through tokens via whitespaces to correctly separate utterances with punctuation e.g., "$13.99". "Token_pos module deals with homograph (words spelt alike but meaning differ) disambiguation and non-standard words e.g., DVD by assignment of token_pos feature to each token via disambiguators consisting of a regular expression and CART tree, ensuring regular expression matching token CART tree is applied and resulting class is correctly assigned to the token via token_pos feature" (Black et al. 2014, chapter 16). Token module deals with analysing tokens into list of words (atom given pronunciation via lexicon), to translate into words and be further labelled with extra tags to identify their type. POS module involves assigning POS tags using HMM (Hidden Markov Model) (trained on hand-labelled data) taggers to words through probability distribution of tags provided a word to disambiguate homographs as POS is only way to choose valid pronunciation of words e.g., "present".

2.3 Pronunciation

Pronunciation dictionaries are used to map words into phonetic transcriptions e.g., CMU, CUVOALD which festival uses. Word module handles lexical lookup/LTS rules involving providing words pronunciations. PostLex module handles the case where natural pronunciation of word isn't found separately from its context. Pronouncing of unknown words, letter sequences involving them converted into sequence of phones known as grapheme-to-phoneme conversion (G2P). "To pronounce, we map its letter to a phone string by training a classifier e.g., CART tree on the training set where it will generate the likeliest phone" (Jurafsky & Martin 2014, chapter 8.2.3).

2.4 Prosody

Phrasify module used to predict phrase breaks done via CART tree (too difficult for hand-written rules) as spoken sentences have structure meaning where some words join naturally whereas breaks are noticeable in others, thus requiring more info e.g., F0 contour to create natural sounding speech. Pauses module used to predict pauses in utterances e.g., commas indicating places where speaker breaks sentence into phrases. "After predicting where to break the sentence into phrases, we might predict the duration of each segment (p) depending on its identity e.g., /m/ is longer than /p/ as well as in the context spoken" (Jurafsky & Martin 2014, 8.3.1). These modules are applied on prosody, utilising linguistic functions in communication e.g., phrasing, rhythm, emphasis, and intonation, also acoustic correlates which are the things that occur to speech signal such as F0, duration.

2.5 Waveform generation

Wave_Synth module is used for waveform generation, involving two kinds of concatenative synthesis (joining units of speech together), diphone and unit selection synthesis. Diphone synthesis model creates waveform through phones sequences via selecting and joining units from small database of pre-recorded diphones (only 1 diphone recording). "Starting at middle of one phone and ending at middle of other thus concatenating diphones giving smoother joins via changing F0, duration." (Jurafsky & Martin 2014, chapter 8.4)

Unit selection synthesis is like diphone synthesis but differs in few ways, contains multiple diphones copies and has longer duration compared to diphone synthesis (1 diphone copy only). No signal processing is applied to concatenated units whereas prosody is changed via TD-PSOLA (algorithm to modify F0 and duration of speech signals so 2 diphones can be concatenated successfully) in diphone synthesis. In a diphone system, such manipulation is essential due to only single recording of each diphone but optional in a unit selection system. Festival voice (uses unit selection) used is configured to not perform any manipulation, rather relying on finding sufficiently well-matched units in the database, not found always due to small database for this voice. Some manipulation would be good to smooth the joins. Commercial unit selection systems use larger databases to increase chances of finding well-matched units. They also use an amount of manipulation to smooth the joins and/or to modify the prosody.

3. Finding and explaining mistakes

3.1 Text normalization

Input provided was “On May 5 1996 the university bought 1996 computers.”. The error is festival misclassifies the ‘1996’ part occurring near the input end as a year (nineteen ninety-six) instead it should be an amount (one thousand nine hundred and ninety-six). Error occurs at token relation of the pipeline, originates after tokenising the input at the detection of non-standard words as token_pos treats it as a year rather than a number because the POS tagger doesn’t take the context into account. Error is quite severe and will affect many utterances of this kind. Short term solution is to insert comma(s) at suitable places to distinguish these kinds of utterances from a year, but a long-term solution is taking context into account when tokenising the input by further labelling with extra tags to help identify their type.

3.2 POS tagging/homographs

Input given was “The produce the farm managed to produce”. Error is festival treats the first word ‘produce’ in the sentence as a vbp (verb non-3rd person singular form) phrase and the second as a vb (verb) phrase whereas the first one should be a noun. Error occurs at the token_pos stage, when assigning POS tags during the tokenisation of speech as festival assigns the wrong tag because of not taking context into consideration. Error is severe as this issue happens with most homographs that don’t have many different pronunciations in its dictionary. It can be resolved by taking the context into account when assigning tags.

name	pos
The	dt
produce	vbp
the	dt
farm	nn
managed	vbd
to	to
produce	vb

Table 1: shows POS tags given to each word in input.

3.3 Phase break prediction

Input given was “my kitten… he’s gone”. Error is that a phrase break is only predicted at the end of the input, a phrase break should be predicted after “kitten” as ellipse indicates a pause. Error occurs at the phrase stage (phrasify) after tokenisation, building token-to-word rules and assigning POS tags, due to error made by CART tree where it can’t predict a phrase break for eclipse. Error is severe as this will happen for inputs of this type; it can be resolved by adding in the eclipse punctuation in the CART tree so it can predict a phrase break for these kinds of inputs.

3.4 Pronunciation

Input given was “athleisure”. Error is that it can’t pronounce the word ‘athleisure’ correctly at all. Error occurs at the word stage after tokenising, adding POS tags and predicting phrase breaks when performing lexical lookup as the phonetic string produced by the LTS module (dictionary + LTS model) is incorrect due to no pronunciation entry existing in the lexicon being selected as different parts of speech are not taken in account although the input was correct. Error is severe and it would affect utterances that have not been included in the dictionary when it was originally written, it can be resolved by adding a pronunciation for these kind of words with its part of speech to the lexicon.

3.5 Waveform generation

Input given was “I have a flight to the U.S.”. Error is that there is a missing diphone [dh] in the text which is an interword “to the”, so silence is inserted instead which sounds bad when played. Error occurs when synthesising the waveform of utterance (Wave_Synth), figure 2.

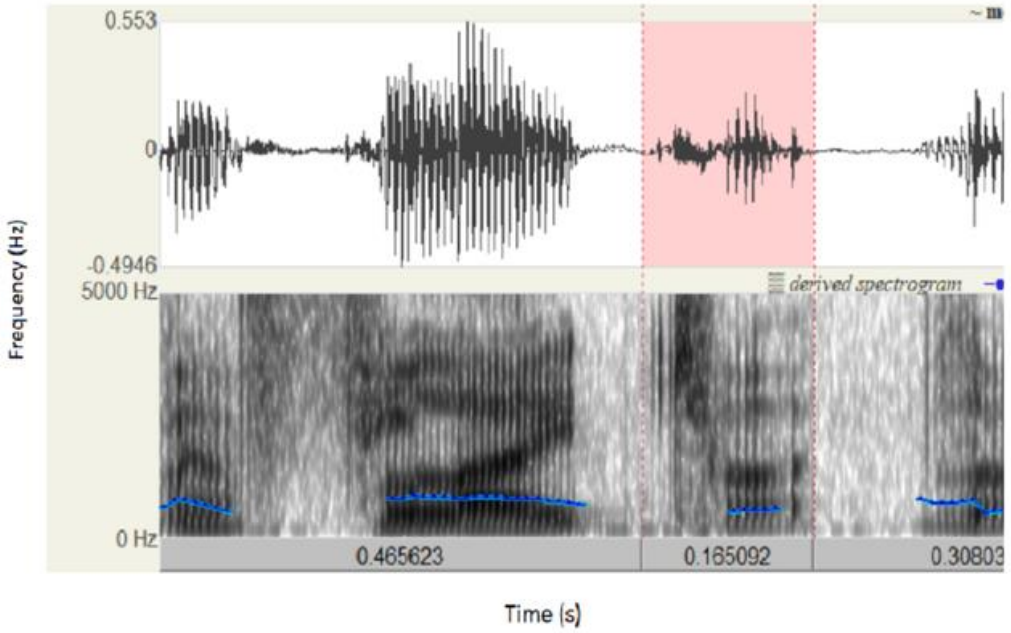

Figure 2: Spectrogram of “I have a flight to the U.S.”, annotated with missing diphone, pitch marks.

Process that led to this error is tokenising the input, identifying tokens, adding POS tags, phrase break and pauses prediction, segment modification via speech. Error is quite severe as this voice has many missing diphones due to the diphone coverage being determined by CMUlex dictionary and the voice built on a different dictionary (Unisyn) and a larger and carefully designed recording script won’t have this issue. To fix this, use the utterance which has a diphone for that utterance as well as ensuring that the voice and diphone are determined by the same dictionary.

For other error, input given was “You have to lead the way”. Error is that festival mispronounces the word ‘lead’ as /lɛd/ instead of /lid/ that is a synonym ‘to direct’. The error doesn’t occur anywhere in the tokenisation, identifying tokens, adding POS tags, phrase break and pauses prediction, segment modification via speech stages as the pronunciation is correct, word is in dictionary with correct POS tags but incorrect pronunciation is retrieved from the dictionary, thus error occurs in Unit stage as the source utterance for the input is different compared to the source of utterance of only ‘lead’ which then causes issues when generating the waveform due to synthetic speech not matching the pronunciation given in the Segment relation, figure 3.

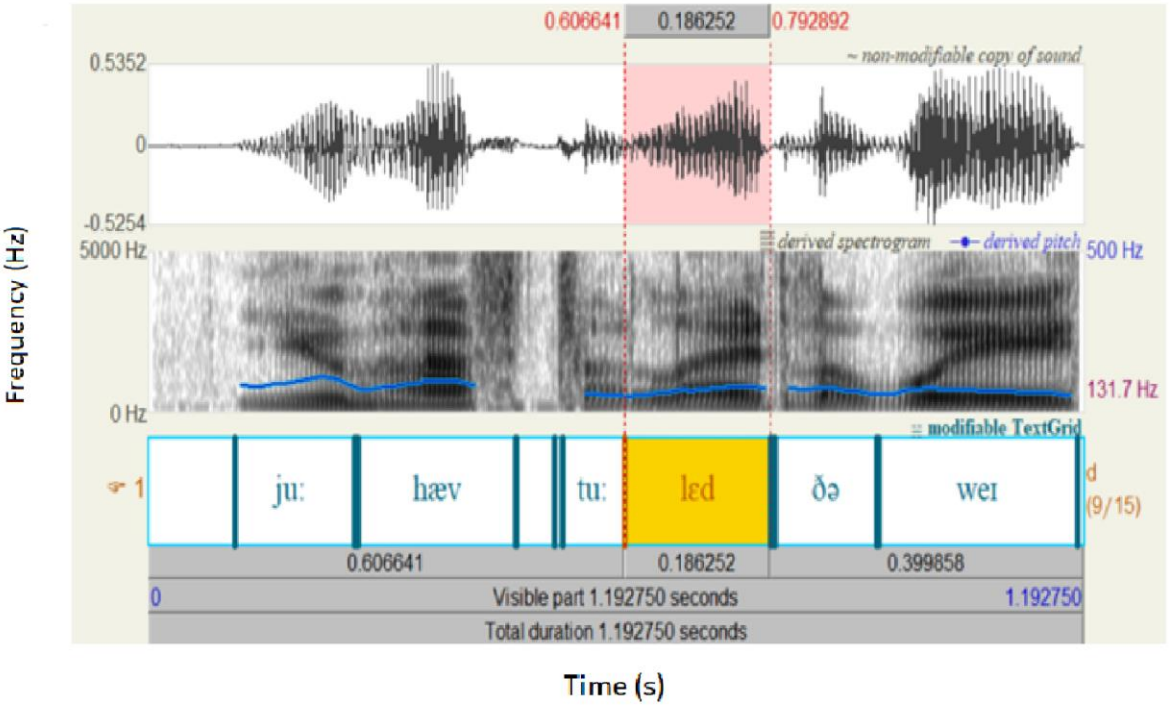

Figure 3: Spectrogram of “You have to lead the way”, annotated with pronunciation, pitch rise area & pitch marks.

Error is quite severe as it leads to mispronunciation and can occur with other utterances e.g., read. It can be resolved by using the same dictionaries since different dictionaries, causes differences in the phones depending on which phonetic transcription is used to determine the phone label alignment (i.e., automatically generated phone boundary times).

3.6 Other types of mistakes

The input was “I love to fly”. Error is that there is rise in the pitch near the end of the utterance making it sound unnatural. Error occurs in the front-end pipeline as no mistakes are made during the tokenisation, identifying tokens, adding POS tags, phrase break and pauses prediction, segment modification via speech stages rather in the intonation stage as the voice used in assignment doesn’t specify anything intonation wise (besides phrase breaks), so the waveform generation module won’t distinguish between rising and falling pitch contours which can be seen in the spectrogram (figure 4). Error might be severe due to festival not being able to differentiate between rising and falling pitch contours leading to

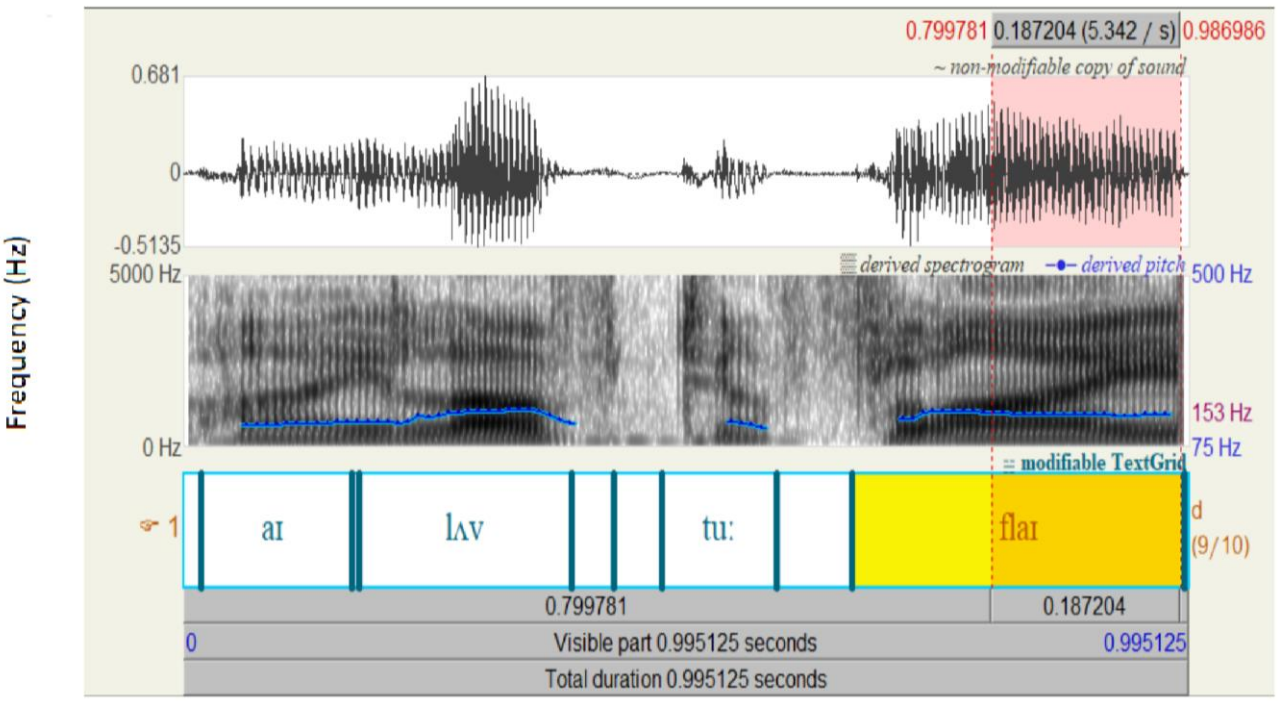

Figure 4: Spectrogram of “I love to fly”, annotated with pronunciation, pitch rise area & pitch marks.

slightly different pronunciations occurring. Error can be fixed in front-end of pipeline by implementing intonation predication in the unit selection voice being used as this would ensure correct prediction of accents on a per syllable basis, so the utterance won’t sound unnatural.

4. Discussion and conclusion

Phrase break, pronunciation errors make Festival sound most unnatural. Text normalisation error would be easy to fix whereas phrase break errors would be most difficult to fix. Front-end errors affect more as waveform generation (back-end) is based on linguistic specification generated in front-end. Festival is good at allowing adding new models, effectively and easily to enhance development, generally bad at pronouncing homograph words and doesn't take context into account when pronouncing.

5. Bibliography

Black, A.W., Taylor, P. and Caley, R. (2014) *The festival speech synthesis system*, *Festival Speech Synthesis System: Table of Contents*. Available at: http://festvox.org/docs/manual-2.4.0/festival_toc.html

Dan Jurafsky and James H. Martin "Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition", 2009, Pearson Prentice Hall, Upper Saddle River, N.J., Second edition, ISBN 0135041961

Hood, M. (2004) *Creating a voice for festival speech synthesis system.* Available at: https://www.cs.ru.ac.za/research/groups/vrsig/pastprojects/049speechsynthesis/paper04.pdf.