Data pruning and objective assessment of intelligibility using confidence measures for unit selection synthesis system

Tejas Godambe, Sai Krishna Rallabandi, Suryakanth V Gangashetty
Speech and Vision Lab, International Institute of Information Technology, Hyderabad, India
{tejas.godambe, saikrishna.r}@research.iiit.ac.in, svg@iiit.ac.in

Abstract

In this paper, we show that improvement in the intelligibility of unit selection synthetic voice built using single speaker audio (not specifically recorded for building a text-to-speech (TTS) system) and which is often without corresponding transcriptions can be achieved using data pruning with the help of two confidence features, (1) combination of phone posterior probabilities obtained from an automatic speech recognition (ASR) system and a multi-layer perceptron (MLP), and (2) unit duration. Posterior probability helps detecting mislabeled and bad acoustic regions in the audio, while unit durational measure helps detecting units having unnaturally short or long durations. Semantically unpredictable sentences from the 2011 Blizzard Challenge data are synthesized and intelligibility is measured in terms of word error rates through subjective listening tests. An objective assessment of intelligibility of synthesized speech (which can be an inexpensive and reproducible alternative to subjective listeners tests) is also performed by passing the TTS signal through ASR system and MLP and classifying each word in the input text as intelligible or unintelligible based on the posterior probability values. The results show that we are able to get a high correlation between subjective and objective intelligibility scores.

Index Terms: Speech synthesis, unit selection system, data pruning, confidence measures, audiobooks, objective measures, speech intelligibility, speech recognition.

1. Introduction

Today, we can record and store large amounts of single speaker audio which could be a lecture speech, spontaneous or read speech on hand-held devices such as smartphones etc. Also, we can download voices of our favourite celebrities from websites such as Youtube, SoundCloud etc. Can we then use these audio which could be a lecture speech, spontaneous or read speech to build an expressive unit selection synthetic voice [1]?

The answer is yes, but there are a few problems related to it. First, these data would often not have transcriptions, and manually transcribing the data could be laborious, time-consuming and expensive. First, if we use an automatic speech recognition (ASR) system to transcribe the data, the ASR output would indispensably contain few substitution, insertion and deletion errors. Presence of these errors can cause a different sentence to be synthesized than the target sentence. Third, the audio data may itself contain bad acoustic (poorly articulated, dis-fluent, unintelligible, inaudible, clipped, noisy) regions as the audio data is not particularly recorded for building a text-to-speech (TTS) synthesis system. In this light, we can see that it becomes necessary to correctly identify and remove all mislabeled and acoustically bad data regions in order to prevent them from corrupting a synthesized voice.

In this paper, we use an audiobook downloaded free of cost from Librivox website (http://librivox.org) for building unit selection TTS system. The audiobooks at Librivox are audio versions of text books, available at Project Gutenberg website (http://gutenberg.org), and are rich in prosody. These books are recorded by volunteers in noise-free rooms. They are not recorded by professional speakers in recording studios as they are not specifically prepared to build TTS systems. Hence, though the audiobooks are largely clean and match with the text, they still contain problems such as dis-fluencies and mismatches between speech and text introduced by the reader. All the above mentioned characteristics of audiobooks available at Librivox website make them representative of the real-world audio (such as lecture speech, spontaneous or read speech) we would record or download from Youtube, SoundCloud etc, which makes them suitable for our current study.

Figure 1: Architectural block diagram of the complete system

In this paper, we specifically focus on the data pruning and objective assessment module in figure[1]. The rest of the paper is organized as follows. Second 2 gives details of the data used to build ASR system and the audiobook data used for building the voice. Section 3 gives an overview of development of ASR, MLP and TTS systems. Section 4 describes the data pruning using confidence measures in detail. Section 5 discusses the objective assessment of intelligibility of synthetic speech. Section 6 covers the evaluation and results. We conclude the paper in section 7.
2. Data preparation

For building the ASR system, we used the Librispeech data. Librispeech is a fairly recently made available continuous speech corpus in English language, which is prepared by collating parts of several audiobooks available at Librivox website. It contains two parts: 460 hours of clean speech (which we used for decoding the audiobook), and 500 hrs of speech data containing artificially added noise.

The audiobook used for building the voice is downloaded from Librivox website. The details of the audiobook are as given in Table 1. We checked that the audiobook is not a part of the 460 hrs corpus, that the reader’s voice quality and speech intelligibility is good, and that there is no noise in the background. Each audiobook contained several 128 kbps mp3 audio files each of which corresponded to a separate chapter. The audio files were downloaded and converted to 16 kHz WAV format. These wave files were then chopped based on silence intervals of 0.3 seconds or more to create phrasal chunks averaging 15 seconds in length. The chunks were power-normalized. An average chunk length of 15 seconds is sufficient enough to capture intonation variation, and does not create memory shortage problems during Viterbi decoding. It is also observed that decoding is faster and more accurate compared to when it is performed for much longer chunks.

Table 1: Details of the audiobook used for building unit selection voice.

<table>
<thead>
<tr>
<th>Name of the audiobook</th>
<th>Author Read by</th>
<th>Running time</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Boats of the Glen Carrig</td>
<td>Hodgson Mills</td>
<td>04:49:20</td>
</tr>
</tbody>
</table>

3. ASR, MLP and TTS system

A context-dependent DNN-HMM acoustic model with p-norm nonlinearities [2] is trained using 460 hours of clean speech (union of train-clean-100 and train-clean-360) subsets from the Librispeech corpus on the top of fMLLR features using the shell script (“egs/librispeech/run.sh”) in Kaldi toolkit [3]. A lexicon containing 200,000 words (released with Kaldi shell scripts for Librispeech corpus) is used. Decoding of the audiobook using above trained acoustic model is done in two passes.

A 3-gram language model (LM)² pruned with threshold 3 x 10⁻⁷ is used for first pass decoding. Then, 4-gram LM is used for lattice rescoring. Even though we require phone labels for the audiobook for building our TTS system, direct phone decoding is not performed as it leads to high errors. Rather, word decoding is performed first, and then word lattices are converted to phone lattices using the lexicon lookup.

An MLP is trained using the 460 hours of Librispeech data, and adapted using the audiobook data and labels hypothesized by the above ASR system. 39 dimensional Mel frequency cepstral coefficients (MFCCs) (including deltas and double-deltas) are used as input features, while 40 nodes corresponding to the 40 English phones comprise the output layer. The architecture of the MLP used is 39L 120N 13L 120N 40M where N and M indicate tangent and soft-max activation functions respectively. The audiobook (after adaptation) is decoded to produce a phone posterior vector for each input frame. For each phone hypothesized by the ASR system, we average its posterior probability with that obtained from the MLP (for the particular phone by averaging over duration hypothesized by the ASR system).

For synthesis, we made a few modifications to the TTS system submitted to Blizzard challenge 2015 [4]. Quinphones and backoff units such as quadruphones, tripophones, biphones and monophones are used as units. The above ASR system produced phones with word-position labels ('b'–word beginning, 'i'–word internal, 'e'–word ending, 's'–word with single phone) which we use for pre-clustering the units. Linear weighted combination of log energy, F0 (along with four context frames [4,5]) and MFCCs forms the concatenation cost. Unit duration is used as the target cost. Viterbi algorithm is used to find the sequence of units that gives the least of the sum total of target and concatenation costs.

4. Data pruning using confidence measures

Data pruning, in the case of TTS systems, involves removing spurious units (which may be a result of mislabeling or bad acoustics) and units that are redundant in terms of prosodic and phonetic features. Pruning spurious units improves TTS output [6] while pruning redundant units reduces database size thus enabling portability [7] and real-time concatenative synthesis [8].

Decision trees have been used to cluster units, and identify and remove units which lie far away from the cluster centers [9]. Confidence features such as log-likelihood ratio [10], transcription confidence ratio [11], generalized posterior probability [12] have also been used for pruning. We use unit duration [13] and average of posterior probability values obtained from the ASR system [12,14] and a multi-layer perceptron (MLP), as confidence features. Posterior probability helps detecting mislabeled and bad acoustic regions, while unit durational measure helps detecting unnaturally short or long units (which may have high posterior probability values) but can make words unintelligible or sound hyper-articulated respectively. Thus both these confidence features are directly related to, and as we will see in section 5, they directly affect intelligibility of speech. The ASR system and MLP both use different models and mechanisms to compute posterior probability and hence their combination may provide complementary information.

In an ASR system, the posterior probability of a phone or a word hypothesis w given a sequence of acoustic feature vectors $O_T = O_1 O_2 ... O_T$ is computed as (given in equation 1) as the sum of posterior probabilities of all paths passing through w (in around same time region) in the lattice. It is computed using forward-backward algorithm over the lattice. $W_s$ and $W_e$ respectively indicate sequence of words preceding and succeeding w in a path in the lattice.

$$p(w|O_T) = \sum_{W_s} \sum_{W_e} p(W_s w W_e | O_T)$$  \hspace{1cm} (1)

Other motivations to use posterior probability as confidence feature are: (1) Posterior probability, by definition, tells the correctness or confidence of a classification, (2) it has been shown to work consistently better than other two formulations of confidence measures, which are confidence measure as a combination of predictor features, and confidence measures posed as an utterance verification problem [15], and (3) posterior probability becomes more reliable when robust acoustic [8] and language models are used [16].
5. Objective assessment of intelligibility of synthetic speech

Subjective listening test for measuring intelligibility of TTS system outputs is expensive, time-consuming and generally irreproducible. Hence, it is desirable to have a mechanism for objective assessment that predicts subjective intelligibility scores quite well. Earlier works verified whether the phone- or word-level content of the TTS signal matches with reference transcription or reference human speech recording. In [17], the TTS output was decoded by an ASR system, and the generated phone graphs were compared to multiple templates of individual phones in varied contexts. Thus, to predict objective intelligibility score for a TTS signal, their system required corresponding reference phone transcription. In [18], objective intelligibility score of a TTS signal was predicted by comparing corresponding phone posterior sequence obtained from an MLP (trained to map MFCCs to phone posterior sequences) with a phone posterior sequence (obtained from the same MLP) corresponding to human reference speech. They used dynamic time warping method for comparison of posterior sequences and Kullback-Leibler (KL) divergence as the local score. Thus, to predict objective intelligibility score for a TTS signal, their system required corresponding reference human speech. In a recent work [19], it was shown that the need for a reference human speech can be alleviated with the use of a Kullback-Leibler hidden Markov model (KL-HMM) trained over the reference human speech. They used an utterance verification approach to classify each word in the input text as intelligible or unintelligible.

In this paper, we use the average of posterior probabilities obtained from the ASR system and MLP for the input TTS signal to decide whether each word in the reference text is intelligible in the TTS signal or not. Unlike [19], (1) we don’t require to train a separate model of human reference speech such as KL-HMM as we already have and can use our trained ASR system and MLP to verify word-level content in the TTS signal, (2) we don’t use KL-HMM scores, but posterior probabilities which are already normalized between [0,1].

6. Evaluation and Results

Subjective listening test is performed to measure the impact of data pruning on intelligibility. Twenty six semantically unpredictable sentences (SUS) [20] are taken from 2011 Blizzard Challenge data [21] for synthesis. Words present in SUS which are infrequent in normal usage and hence may not be known to the listeners are replaced by easier words. Twelve graduate students participated in the listening test. Twelve utterances are synthesized for each SUS sentence using units corresponding to different cells, i.e. different posterior probability and duration ranges, in Table 2. The 12 utterances are played out to 12 different listeners to 26 different sentences. Units are expected to different combinations of posterior probability and duration zscore.

Table 2: Average subjective intelligibility scores in WER for different combinations of posterior probability and duration zscore.

<table>
<thead>
<tr>
<th>Posterior probability / zscore</th>
<th>&lt;2.5</th>
<th>-2.5 - 0</th>
<th>0 - 2.5</th>
<th>&gt;2.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0 - 0.5</td>
<td>35.3</td>
<td>31.7</td>
<td>25.6</td>
<td>16.3</td>
</tr>
<tr>
<td>0.5 - 0.9</td>
<td>32.2</td>
<td>28.8</td>
<td>16.4</td>
<td>10.2</td>
</tr>
<tr>
<td>&gt;0.9</td>
<td>25.9</td>
<td>20.1</td>
<td>10.6</td>
<td>5.5</td>
</tr>
</tbody>
</table>

duration zscore for each unit is computed using the equation

\[
\text{duration zscore} = \frac{\text{duration} - \text{mean}}{\text{standard deviation}}
\]

Table 2 shows average subjective intelligibility scores (in word error rate (WER)) for different ranges of posterior probability and duration zscore. Steps of 2.5 duration zscore are inspired from [13] which showed that 50% of synthesized sentences were affected by removing units over this threshold. The

The TTS signal is decoded using the ASR system and MLP, and average of posterior probabilities is taken. Now, for each word in the input text we have a posterior probability value. We should classify words above a threshold as intelligible and others as unintelligible. The task is to find the optimal value of threshold that gives least difference in subjective and objective intelligibility scores. For this, utterances corresponding to two out of six sentences are used as development set. Figure 2 shows the difference in subjective and objective intelligibility scores (DSOIS) computed for different threshold values. We find that DSOIS is high for low threshold values as many words (which are actually unintelligible) are classified as intelligible by the objective assessment module. DSOIS is also slight high for high threshold values where objective assessment module rejects a few words terming them as unintelligible when, in actual, they are intelligible. Minimum DSOIS value is found for threshold equal to 0.7.

Now, the DSOIS is computed over the test set (utterances corresponding to remaining four sentences) using the threshold of 0.7. Table 3 summarizes DSOIS for development and test

Table 3: Preference test results.

<table>
<thead>
<tr>
<th>TTS sample set</th>
<th>Preference (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>72</td>
</tr>
<tr>
<td>Y</td>
<td>28</td>
</tr>
</tbody>
</table>
sets. The table shows that there is a 95% and 92% correlation between subjective and objective intelligibility scores for development and test sets which is reliable enough to proceed with carrying out an intelligibility assessment of the voice on a larger representative text corpus.

Table 4: DSOIS for development and test sets.

<table>
<thead>
<tr>
<th>Set</th>
<th>DSOIS (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>dev</td>
<td>5</td>
</tr>
<tr>
<td>test</td>
<td>8</td>
</tr>
</tbody>
</table>

7. Conclusions

In this paper, we showed that improvement in the intelligibility of unit selection synthetic voice built using single speaker audio (not specifically recorded for building a text-to-speech (TTS) system and which is often without corresponding transcriptions) can be achieved using data pruning with the help of two confidence features, (1) combination of phone posterior probabilities obtained from an automatic speech recognition (ASR) system and a multi-layer perceptron (MLP), and (2) unit duration. Semantically unpredictable sentences from the 2011 Blizzard Challenge data were synthesized and intelligibility was measured in terms of word error rates through subjective listening tests. An objective assessment of intelligibility of synthesized speech was also performed. The results showed that we are able to get a high correlation between subjective and objective intelligibility scores.

8. References