Towards Improving Intelligibility of Black-Box Speech Synthesizers in Noise

Thomas Manzini[1] and Alan Black[1]

Carnegie Mellon University, Pittsburgh PA, 15213, USA {tmanzini,awb}@cs.cmu.edu

Abstract. This paper explores how different synthetic speech systems can be understood in a noisy environment that resembles radio noise. This work is motivated by a need for intelligible speech in noisy environments such as emergency response and disaster notification. We discuss prior work done on listening tasks as well as speech in noise. We analyze three different speech synthesizers in three different noise settings. We measure quantitatively the intelligibility of each synthesizer in each noise setting based on human performance on a listening task. Finally, treating the synthesizer and its generated audio as a black box, we present how word level and sentence level input choices can lead to increased or decreased listener error rates for synthesized speech.

Keywords: Speech · Synthesized · Noise · Radio · Intelligibility

1 Introduction

Synthetic speech systems have undergone a great deal of research in the past years. Other research efforts have attempted to predict the intelligibility of different synthesizers in different settings [16, 17]. However, to the author's best knowledge, all work in this area has been done from the perspective of improving synthesized audio [9, 11], rather than the synthesizer inputs themselves.

This paper aims to determine if intelligibility can be predicted from the content fed to the synthesizer. In this work we explore how to predict if certain words and sentences will be understood by users and how these predictions can be used to formulate or reformulate a sentence for speech in a noisy environment. This is done by treating the synthesizer as a black box and measuring only the inputs and the outputs.

Our work is specifically motivated by automated disaster response. Much work has been done using artificial intelligence to handle emergency and disaster situations [8, 7]. The integration of speech is a necessary and natural expansion of this research. We foresee speech systems needing to operate in noisy environments where synthetic speech may be broadcast over a radio frequency or near rescue equipment. Both present multiple different issues regarding types of noise. In this work, we use the noisy environment of a radio channel as a test bed for intelligibility.

2 Related Work

Several works have explored multiple different types of noise and multiple different types of speech [16, 17]. There are two relevant concepts within the field that are at play in this work: the field of speech in noise and the work surrounding listening tests.

2.1 Speech In Noise

The intelligibility of speech in noise is the measure of how well audio - containing either natural or synthetic speech - can be understood in a noisy environment. A noisy environment can range from the chatter of a restaurant [16] to the sounds of helicopters and the battlefield [18]. In all of these environments a listener may confuse or misinterpret speech because of noise.

In past works, authors have shown several key concepts. First, when measuring the kinds of errors list listeners make, [12] has shown that while keyword error rate (KER) may be a more accurate measure, simple word error rate (WER) follows KER closely and is less time intensive to calculate. As such, we use WER for our measurements. At the same time, [19] has shown that there are instances where there are disparities between WER and other metrics, such as concept error rate. We see in our data that WER tends to follow concept error rate.

2.2 Listening Tests

Listening tests are a common way to evaluate the intelligibility of a voice [10, 13, 15]. Compared to automated methods and metrics, human evaluation is traditionally regarded as the most effective method for evaluation. As a result Listening tests have been used to evaluate synthesizers and intelligibility of both synthetic and natural speech in noise.

3 Experimentation

We explore the effect of three different types of noise on three different synthesizers. This is in an attempt to understand the how humans understand different synthesizers generally, as opposed to possibly overfitting to one synthesizer or one noise setting.

3.1 Structure

We asked English speaking listeners who over eighteen years old to transcribe audio from a series of thirty different audio files. These audio files were generated by selecting random sentences from the *Smart-Home* dataset [14] and having one of three different synthesizers generate an associated audio file then one of three different noise levels was applied to the audio. The result was captured and stored for the listening task. This was done thirty times for each listening task, resulting in thirty unique audio files for each listener per task. While our research is motivated by emergency/disaster response use case, we selected this dataset because it has a demonstrably diverse vocabulary that would, in theory, lend itself well to determining the intelligibility of various words. During the initial stages of this research we did explore other datasets, including one of radio traffic, between emergency medical services (EMS) Personnel and their dispatch center, but found that it was not lexically diverse enough for the purposes of this research.

Two different types of listening tasks were performed: one to generate training data and another for testing data. In the training task forty-five listeners listened to 450 different audio files. Each file was labeled by three different listeners. In the testing task a fifty listened to 150 different audio files. Each file was labeled by ten different listeners. These two different tasks were done so that the test data would be the most representative of the behavior of users in a noisy environment.

3.2 Synthesizers

We used three different synthesizers for our experimentation: the E-Speak Synthesizer [5], the Flite Synthesizer [1], and the Google Synthesizer [6]. All synthesizers used an English-speaking male voice, but these three synthesizers each have their own specific settings.

E-Speak. We used the E-Speak Synthesizer with primarily the default settings. We specified two unique settings when generating our sound files: the use of the *en* voice which corresponds to an English speaking male and the use of a voice speed of level 120 (down from 175). This was done to better align the speeds of the voices of the different synthesizers.

Flite. We used the CMU Flite (Festival Lite) synthesizer with the default settings. We specified that the synthesizer must use the $cmu\_us\_eey.flitevox$ voice that came prepackaged with the standard release of Flite.

Google. We used the Google Text to Speech system defined within the Python gTTS module. We specified that the synthesizer must use the $en - us$ voice that came with the release of gTTS. All other settings were left at default values.

3.3 Noise

We used three different noise levels each consisting of three different filters applied at different values. First we impose an ambient noise filter designed to replicate radio static. For this filter, we take the original sound and at each time step sample a value from a random normal distribution centered at the original sound. The standard deviation of this normal distribution was varied at each noise level. Next we perform a low pass filter with a variable threshold. Finally we perform and high pass filter with a variable threshold. Varying the parameters to these three different filters provide several knobs we can turn to increase or decrease the noise within the audio files. We make no claim about how well these different filters replicate the noise present on a radio channel, as that can vary based on the radio manufacturer, the frequency used, and the type of system in use. We only state that this noise is subjectively similar to that of an active radio channel. Further work would be required to determine the best noise filters needed to replicate each specific radio channel.

We chose three different noise settings that would be presented to users. These noise settings were not intended to be ranked by difficulty, but were intended to represent three distinct kinds of noise that could cause a listener to make transcription errors. We believe that the reasons why certain noise settings are more likely to cause listening errors are out of scope of this work and could be the subject for further research.

Noise Level 1. Random noise filter standard deviation: 0.3; Low pass frequency cutoff: 300.0 Hz; High pass frequency cutoff: 2500.0.

Noise Level 2. Random noise filter standard deviation: 0.4; Low pass frequency cutoff: 400.0 Hz; High pass frequency cutoff: 2000.0.

Noise Level 3. Random noise filter standard deviation: 0.5; Low pass frequency cutoff: 500.0 Hz; High pass frequency cutoff: 1500.0.

4 Listening Test Results

we presented users with different audio files and recorded their precision/word error rate. We include the complete breakdown of user performance below in Table 1. For all experiments we segment the data based on both synthesizer and noise level. We collected approximately fifty different sentences at each different noise level and synthesizer combination for training and approximately sixteen different sentences at each noise level for testing.

Transcription Precision Score	Noise Level 1	Noise Level 2	Noise Level 3	Average
Espeak	0.227	0.196	0.242	0.222
Flite	0.346	0.375	0.343	0.355
Google	0.542	0.639	0.559	0.580
Average	0.372	0.403	0.381	

Table 1. Precision (1.0 - WER) of Word Level Transcription per noise and synthesizer on the training data.

4.1 Listening Test Results Discussion

Table 1 demonstrates the word error rates of the different synthesizers and noise levels. We see a clear trend among the different synthesizers that the highest performing synthesizer was the Google synthesizer, followed by Flite, and finally by ESpeak. These results indicate that the quality of the synthesizer plays a role in its intelligibility in noise.

Empirically, it appears that Noise Setting 2 is the most intelligible noise setting, followed by Setting 3, and then Setting 1. It should be noted that results for settings 1 and 3 are very similar and differ somewhat. At the same time there does appear to be some variation between the different synthesizers. While Noise Setting 2 is the least intelligible for Espeak, it is the most intelligible for Google. From this observation, we can conclude that the intelligibility of a given synthesizer in a given noise setting depends primarily on the synthesizer and not on the noise setting. We make no claim regarding why certain noise settings are more intelligible than others, we believe this to be an avenue of further research.

5 Predictive Results

We make intelligibility predictions at the sentence level and the word level. At the sentence level, a model could estimate which paraphrasings are most likely to be understood by listeners. At the word level a model could rank synonyms of specific words so that they are more likely to be understood. At both levels of granularity we explore the application of point-wise and pair-wise ranking for estimating intelligibility. While list-wise reranking is an obvious extension to this work, we do not have enough sentence level, or word level data to make list-wise reranking models feasible. We present the results of this predictive exercise below.

Work in this field often uses metrics such as the DAU metric [3] or the Glimpse proportion measure [2] to attempt to model intelligibility. These metrics are based off the audio features of your synthesizer. Since we attempt to predict intelligibility based on non-audio features these metrics are out of the scope of this work.

5.1 Sentence Level Intelligibility Prediction

At the sentence level, we try to determine if one sentence is more intelligible than another. We explore this in two ways: first, we trained a machine learning model to estimate the average word error rate of a given sentence, and second, we trained a pair-wise reranking model to attempt to determine if one particular sentence is more intelligible than another.

Sentence Level Word Error Rate Estimation At the sentence level we attempt to train a machine learning model to predict the average word error rate of a given sentence. In order to do this, we construct a feature vector that contains a number of sentence level features. We trained a simple linear model with sigmoid activation. We found that we achieved the best results when using simple models. We used several different features to estimate the word error rate of a sentence but few were effective given the low amount of data. Our features included average word rank, average word length, sentence length, word count, and percent of unique characters. We define word rank as the ranked position of a term, based on how frequently that term appears in the Corpus of

Contemporary American English [4]. We define word length as the length of a particular word in characters. We define average word rank and average word length as the average of these respective values. Other features were explored but eventually discarded. The results of this model on the test set are presented in Table 2.

Point Wise Reranking - Sentences			
Synthesizer	Noise Level	MSE (Test)	Spearman's R (Test)
Espeak	1	0.0227	0.2258
Espeak	2	0.0256	-0.3728
Espeak	3	0.0311	-0.4650
Flite	1	0.0477	-0.1225
Flite	2	0.0451	0.4621
Flite	3	0.0587	-0.1863
Google	1	0.0505	0.1176
Google	2	0.0915	-0.2943
Google	3	0.0701	0.0662

Table 2. Performance of our linear error estimator for sentence level error estimation.

Pair-Wise Sentence Reranking We constructed a linear model with tanh activation to estimate which sentence is the most intelligible. We do this by feeding one feature representation of each sentence into the linear model. The model then estimate if the first sentence is more intelligible (labeled +1), the second sentence is more intelligible (labeled -1), or if the intelligibility of the sentences are equal (labeled 0). We then train this linear model and evaluate it on the test set. The results of this evaluation are presented in Table 3.

Pair Wise Reranking - Sentence			
Synthesizer	Noise Level	MSE (Test)	MSE Variance (Test)
Espeak	1	0.3441	0.1211
Espeak	2	0.5960	0.1217
Espeak	3	0.4641	0.2591
Flite	1	0.7074	0.1496
Flite	2	0.3979	0.351
Flite	3	0.6095	0.3688
Google	1	0.6679	0.1638
Google	2	0.4076	0.1440
Google	3	0.5080	0.1966

Table 3. Performance of our linear pair-wise sentence level reranking model.

5.2 Word Level Intelligibility Prediction

At the sentence level we are attempting to estimate which words would be most intelligible, either on their own, or when compared to another word. For use in a real world setting the models discussed here could be used to estimate the intelligibility of synonyms of different words in a sentence so as to maximize the intelligibility of a sentence overall. As a result, estimating which words are going to be the most intelligible is an obvious initial step to estimating the overall intelligibility of a sentence, phrase, or other unit of speech.

Word Error Rate Estimation Working at the word level we have access to significantly more data. Here we trained a machine learning model to attempt to estimate the WER of a particular word in a given sentence. This is different from the sentence level task of the same name because we have features for the word, but also features for the context of the word (eg. the surrounding words).

We construct a linear model with sigmoid activation to attempt estimate the error. We used several different word level features regarding the words themselves, and their surrounding contexts. Our word level features included: word rank, percent of vowels in the word, percent of consonants in the word, length of the word, and the percent of unique characters in the word. We define word rank in the same manner described in section 5.1. Our context level features included: the same word level features for both the previous and next word, the length of total number of words in the sentence, and the number of total unique words in the sentence. The results of this evaluation can be found in Table 4.

Point Wise Reranking - Words			
Synthesizer	Noise Level	MSE (Test)	Spearman's R (Test)
Espeak	1	0.0429	-0.2516
Espeak	2	0.0426	0.0676
Espeak	3	0.0318	-0.1120
Flite	1	0.0932	-0.1612
Flite	2	0.1032	0.1346
Flite	3	0.0468	-0.1589
Google	1	0.0902	0.2173
Google	2	0.1182	0.3114
Google	3	0.0801	0.0178

Table 4. Performance of our linear error estimator for word level error estimation.

Pair-Wise Word Reranking To perform pairwise reranking, we changed the layout of our model slightly. We now pass two times the number of features to our model, one for the first word and one for the second word. The word level features that are fed to the model are similar as in the above section, but they have had the features regarding sentence context removed, and contain only features regarding the neighboring words. Like the sentence pair wise reranking schema the model then has to estimate if the first word is more intelligible (labeled +1), the second sentence is more intelligible (labeled -1), or if the intelligibility of the sentences are equal (labeled 0). We trained this linear model and evaluated it on the test set. The results are presented in Table 5.

Pair Wise Reranking - Words			
Synthesizer	Noise Level	MSE (Test)	MSE Variance (Test)
Espeak	1	0.1946	0.1151
Espeak	2	0.1546	0.0901
Espeak	3	0.1610	0.0959
Flite	1	0.1896	0.0972
Flite	2	0.2022	0.1167
Flite	3	0.2009	0.1052
Google	1	0.1783	0.0941
Google	2	0.1794	0.1031
Google	3	0.1840	0.1076

Table 5. Performance of our linear pair-wise word level reranking model.

6 Results Discussion

From our data, we can see that the sentence level error estimation and pair-wise reranking methods are ineffective at the current data scale. For the Spearman's correlation we can see that there is no consistent behavior between the different error models. In the case of the pair-wise reranking for sentences we still see poor performance. Not only is the MSE fairly high for a problem like this, the variance of the MSE is much larger than would be anticipated. We believe that these are problems that could be solved with additional data, but at the moment our models are not capable of performing this task at the sentence level.

For the word level point-wise reranking we can estimate intelligibility for the Google synthesizer to some extent. This is indicated by the Spearman's correlation which is either positive or near zero. However, this is not the case for other synthesizers. The pair-wise word ranking is more stable than the sentence level reranking. For all synthesizers and for all noise levels we see model behavior indicative of estimating the correct word in a reranking context. At the same time, the variances of the MSE are within a reasonable bound and upon closer inspection we do not see a many outliers that could skew these results.

Based on our results and given the data that we have presented here, we find that we are positively able to rerank individual terms based on lexical features to estimate their intelligibility. Our results demonstrate that this methodology works best in the pairwise reranking context for this particular data scale. We believe that additional labeled data will improve performance.

7 Future Work

The most significant piece of future work is more data with intelligibility labels for different noise settings and synthesizers. Additional evaluations on different features and different models for predicting intelligibility on the lexical level would be useful.

8 Conclusion

This work has explored the intelligibility of three different synthesizers in three different noise settings. We have evaluated these synthesizers in these noise settings on a human listening task and we have measured performance along metrics that reflect intelligibility. Further we have explored methods that have shown some predictive power regarding how predictable intelligibility is on a lexical level. We show that even with limited data you are able to rerank words and estimate which word will be more intelligible in a given context.

9 Acknowledgements

We would like to acknowledge several people for their help and support on this work. Particularly Carolyn Penstein, Rajat Kulshreshtha, Abhilasha Ravichander, and the officers of CMU EMS. As well as the several people who helped edit this work, especially Elise Romberger. Finally, thank you to reviewers reading and examining our experiments, methodology, and submission.

References

[1] Alan W Black and Kevin A Lenzo. “Flite: a small fast run-time synthesis engine”. In: *4th ISCA Tutorial and Research Workshop (ITRW) on Speech Synthesis*. 2001.

[2] Martin Cooke. “A glimpsing model of speech perception in noise”. In: *The Journal of the Acoustical Society of America* 119.3 (2006), pp. 1562–1573.

[3] Torsten Dau, Dirk Püschel, and Armin Kohlrausch. “A quantitative model of the ”effective” signal processing in the auditory system. I. Model structure”. In: *The Journal of the Acoustical Society of America* 99.6 (1996), pp. 3615–3622.

[4] Mark Davies. *The corpus of contemporary American English (Coca): 450 million words, 1990-2012*. Brigham Young University, 2002.

[5] Jonathan Duddington. *eSpeak text to speech*. 2012.

[6] Pierre Nicolas Durette. *gTTS: a Python interface for Google’s Text to Speech API*. [Online; accessed April 15th, 2018]. 2017–. URL: `https://github.com/pndurette/gTTS`.

[7] Frank Fiedrich and Paul Burghardt. “Agent-based systems for disaster management”. In: *Communications of the ACM* 50.3 (2007), pp. 41–42.

[8] Muhammad Imran et al. “AIDR: Artificial intelligence for disaster response”. In: *Proceedings of the 23rd International Conference on World Wide Web*. ACM. 2014, pp. 159–162.

[9] Sunil Kamath and Philipos Loizou. “A multi-band spectral subtraction method for enhancing speech corrupted by colored noise.” In: *ICASSP*. Vol. 4. Citeseer. 2002, pp. 44164–44164.

[10] Mead C Killion et al. “Development of a quick speech-in-noise test for measuring signal-to-noise ratio loss in normal-hearing and hearing-impaired listeners”. In: *The Journal of the Acoustical Society of America* 116.4 (2004), pp. 2395–2405.

[11] Robert McAulay and Marilyn Malpass. “Speech enhancement using a soft-decision noise suppression filter”. In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 28.2 (1980), pp. 137–145.

[12] Youngja Park et al. “An empirical analysis of word error rate and keyword error rate”. In: *Ninth Annual Conference of the International Speech Communication Association*. 2008.

[13] M Kathleen Pichora-Fuller, Bruce A Schneider, and Meredyth Daneman. “How young and old adults listen to and remember speech in noise”. In: *The Journal of the Acoustical Society of America* 97.1 (1995), pp. 593–608.

[14] Abhilasha Ravichander et al. “How Would You Say It? Eliciting Lexically Diverse Dialogue for Supervised Semantic Parsing”. In: *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*. 2017, pp. 374–383.

[15] Astrid Schmidt-Nielsen. *Intelligibility and acceptability testing for speech technology*. Tech. rep. NAVAL RESEARCH LAB WASHINGTON DC, 1992.

[16] Cassia Valentini-Botinhao, Junichi Yamagishi, and Simon King. “Can objective measures predict the intelligibility of modified HMM-based synthetic speech in noise?” In: *Twelfth Annual Conference of the International Speech Communication Association*. 2011.

[17] Cassia Valentini-Botinhao, Junichi Yamagishi, and Simon King. “Evaluation of objective measures for intelligibility prediction of HMM-based synthetic speech in noise”. In: *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE. 2011, pp. 5112–5115.

[18] Andrew Varga and Herman JM Steeneken. “Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems”. In: *Speech communication* 12.3 (1993), pp. 247–251.

[19] Ye-Yi Wang, Alex Acero, and Ciprian Chelba. “Is word error rate a good indicator for spoken language understanding accuracy”. In: *Automatic Speech Recognition and Understanding, 2003. ASRU’03. 2003 IEEE Workshop on*. IEEE. 2003, pp. 577–582.