Speech Prosody 2022
23-26 May 2022, Lisbon, Portugal

Incorporating Prosodic Events in Text-to-Speech Synthesis

Rose Sloan[1], *Adaeze Adigwe*[1], *Sahana Mohandoss*[1], *Julia Hirschberg*[1]

[1]Columbia University

rsloan@cs.columbia.edu,aoa2147@columbia.edu,sm4769@columbia.edu,julia@cs.columbia.edu

Abstract

While producing accurate prosody can significantly improve the naturalness and comprehensibility of synthesized speech, many Text-to-Speech (TTS) systems still do not explicitly model prosody. In this paper, we present an approach for incorporating prosodic events, specifically phrase breaks and pitch accents, into TTS output using a two-step pipeline. In the first step, we use a large number of linguistic features to create a model for predicting the locations of prosodic events from text. In the second step, we incorporate these events into the front end of a DNN-based TTS pipeline. We crowd-source labels for pairs of utterances created with and without the new pipeline. Our results show that listeners strongly prefer a voice created using this pipeline to the baseline voice, indicating that this approach of explicitly modeling prosodic events is a fruitful area of research.

Index Terms: text-to-speech, prosody

1. Introduction

The ability to produce appropriate prosody is a crucial component of natural-sounding text-to-speech systems. In particular, accurately modeling prosodic events such as phrase breaks and pitch accents can have an important impact on an utterance's naturalness and even its meaning. However, many modern text-to-speech systems do not model prosody explicitly, relying instead on large training sets and complex models to produce reasonable prosody. Systems that do model prosody also have a number of shortcomings. For example, some systems only model prosody in specific cases (such as emphasizing words in all caps), while other systems only model prosody at the sentence level, providing no way to control low-level prosodic events. While these models can produce high quality speech, they provide no way to capture specific prosodic events or correct unnatural output.

This problem can be especially troubling for domains with long utterances and distinctive prosody, such as the domain of radio newscasts. In radio news, utterances can be much longer than the standard utterances used to train TTS, often consisting of short paragraphs containing 2 to 4 long and complex sentences. In these cases, an inadequate prosody model, particularly a poor phrasing model, can lead to unnatural phrase breaks and speech that is difficult to follow and understand. Therefore, in this work, we decided to focus our work on radio news using the Boston University Radio News Corpus (BURNC), a ToBI-labeled corpus of radio news.

We approached this problem using a two-step process. First, since most corpora are not prosodically labeled, and our system needed to be able to synthesize appropriate prosody for any novel sentence, we created a model for predicting prosodic events from text only, using a number of linguistic features. Specifically, we looked at two binary classification tasks, predicting the location of phrase boundaries and predicting the location of pitch accents. In the second step, we added these predicted prosodic events to the front end of the Merlin text-to-speech system in order to explicitly train our TTS system to synthesize them.

In Section 2 we describe related work. We describe the corpus we train on in Section 3. Section 4 discusses the features we use for prosody prediction. The TTS pipeline we use to produce our synthesized speech is explained in Section 5. Section 6 presents our results using Amazon Mechanical Turkers' judgments. Finally, in Section 7 we conclude and discuss future research.

2. Related Work

Prosody prediction from text has been an active area of research for several decades. Early models used relatively simple features, such as part of speech, a word's position in a sentence or paragraph, and punctuation [1, 2, 3]. Somewhat later work showed that more complex syntactic features, including information about syntactic phrases and supertags, could improve the model [4, 5].

More recent work on developing features for prosody prediction has focused on a combination of linguistic and word embedding features. Obin and Lachantin found that a set of rich syntactic features could be used to predict prosodic events in both read and spontaneous speech [6]. Mishra et al. showed that combining syntactic and semantic features, such as part of speech tags and dependency relations, could provide an effective substitute for lemmatized word identity in predicting prosodic phrase boundaries [7]. Their work also showed that it is important to employ a context window corresponding to the average length of an intonational phrase – five to six words – in phrase boundary prediction. Rendel et al. found that including GloVe and CBOW embedding features could provide a significant 35% relative gain for prediction in a pitch accent task, although only a modest 2.4% gain for phrase boundary detection was achieved [8].

There are several approaches to incorporating prosody into TTS pipelines. Some have taken an approach similar to ours and have incorporated prosodic features directly into the front end of a TTS system. Malisz et al. altered Merlin's front end to include the level of prominence of each word as a feature [9], while Fujimoto et al. included markings for high and low accents into the front end of an end-to-end system for Japanese TTS [10]. While these papers demonstrate that incorporating prosodic features can improve TTS output, unlike our approach, they do not provide a method for generating these features for novel utterances.

Other work has focused on incorporating linguistic features into the front end of either RNN-based or end-to-end systems. Guo et al. demonstrated that including syntactic features in the front end of an end-to-end system can noticeably improve prosodic output [11], while others have shown that including word or character embeddings on the front end of systems can also improve output [12, 13].

10.21437/SpeechProsody.2022-59

Finally, some approaches focus not on modeling low level prosodic events but on creating an prosodic embedding layer that affects the prosody of an entire utterance. The most notable of these approaches is Wang et al.'s Global Style tokens, which use a variational autoencoder to create "style embeddings" that can be used to alter speaker style or transfer style from one utterance to a novel utterance [14]. Tyagi et al. present a similar approach but also provide a way to create style embeddings for novel utterances, using syntactic and word embedding features [15]. While these approaches can create high quality synthesized speech, they do not provide any control over low-level prosodic events and therefore do not provide any way to correct errors in prosodic output.

3. Corpus

For this work, we focused on the Boston University Radio News Corpus (BURNC), a corpus of English language radio news compiled by Mari Ostendorf, Patti Price and Stephanie Shattuck-Hufnagel, with the purpose of creating clean speech data conducive to prosody research [16]. BURNC consists of one portion of read speech recorded in a lab and another, larger portion recorded from actual broadcast news. For the sake of consistency, we use only the latter portion in this work, which consists of over seven hours of data. There are seven speakers in the corpus, three female and four male. The data has been segmented into utterances that are generally the length of short paragraphs, no longer than three or four sentences. Because this is high quality broadcast news data, there are very few disfluencies or other irregularities throughout the data.

BURNC includes a number of useful annotations, including gold standard transcriptions, part of speech tags, and, for a portion of the corpus, ToBI (Tones and Break Indices) labels [17]. The transcriptions and ToBI labels were created manually; the part of speech tags were automatically generated and then hand-corrected. The ToBI-labeled portion of the corpus consists of slightly under one fifth of the total corpus and includes data from five of the seven speakers.

At one point, we did attempt to extend these ToBI labels to the remainder of the corpus using automatic methods. However, unfortunately, we found that, due to inaccuracies and inconsistencies in the automatically generated labels, including these labels did not actually improve our prosodic event model. Therefore, we trained that model, which is the first stage of our pipeline, only on the manually labeled portions of the corpus. However, we trained our TTS model on all utterances from female speakers.

4. Prosodic Event Modeling

As mentioned earlier, we created two binary classification models, one for predicting whether a word is followed by a 4-level ToBI phrase boundary and one for predicting whether a word has a pitch accent. We chose to look only at these binary tasks, as opposed to anything more fine-grained, since having high accuracy models is crucial in the next step; we found that it is difficult to ensure high accuracy in any more specific models. (In fact, when labeling phrase boundaries, even human annotators often struggle to identify 2- and 3-level ToBI boundaries.) These models used a variety of features, all text-based and extractable using easily available NLP tools; this allowed them to be run on novel sentences.

The model used here is very similar to the one presented in [18], where we explored the potential feature set in more detail. There, we found that we could get an accuracy of 93.4% for phrase boundaries and 81.9% for pitch accents. As these accuracy numbers are relatively high, we found that the predictions generated by this model were sufficiently accurate to allow us to train TTS systems using them. These models used a Random Forest classifier trained on a large number of features. A description of the features used in our best models is outlined below.

4.1. Word-Level Features

Our model began with a few simple word-level features. These included the number of syllables in each word and the punctuation following each word. We also included the named entity recognition tag for each word, extracted using the Stanford CoreNLP toolkit [19].

For our pitch accent model, we also included Linguistic Inquiry and Word Count (LIWC) dimensions for each word as features [20]. LIWC is a system for categorizing words, which relies on a dictionary at its heart to define categories, related to both a word's semantic and syntactic properties. For example, the word *cried* belongs to five categories: sadness, negative emotion, overall affect, verbs, and past focus. We used LIWC to assign each word to 73 dimensions of emotions, thinking styles, social concerns, and parts of speech. Notably for our purposes, there is a function word category in LIWC, which, not surprisingly, turned out to be the single most heavily weighted feature in our pitch accent model.

4.2. Syntactic Features

The largest group of features in our model was comprised of syntactic features. Gold standard part of speech tags are included with BURNC; these were used as features. Additionally, we derived various features from the dependency and constituency parses for each sentence, which we extracted using Stanford CoreNLP.

From the dependency parse, we extracted the syntactic function of each word and included it as a feature. From the constituency parse, we extracted a large number of features: First, we included the parse tree width and height for each parse, as well as each word's depth within the tree. Then, we examined the smallest non-trivial constituent containing each word, including its size and root label, as well as the position of the word within this constituent. Next, for each word, we identified the minimal spanning tree between the current word and the next word. We included the size and root label of this tree as features.

We also included each word's supertag, based on prior work indicating the usefulness of supertags to predict prosody [4]. Supertags provide a more fine-grained syntactic tag than part of speech tags. Based on the Tree-Adjoining Grammar formalism, they assign each word a portion of a syntax tree, allowing them to capture more detailed information than a part of speech tag. (For example, a supertag can distinguish between a verb in active voice and one in passive voice.) For our experiments, we extracted supertags using a BiLSTM-based tagger pre-trained on Wall Street Journal data [21].

4.3. Co-Reference Features

It has long been hypothesized that given information is less likely to be accented than new information. Therefore, for our pitch accent model only, we included a handful of features related to a word's previous co-references. We extracted co-

references for the whole corpus using Stanford CoreNLP's coreference resolution tool. Using these groups of co-referents, we then extracted a number of features. These features included the number of prior co-reference mentions for each word, the syntactic function of each word's previous mention, the part of speech of each word's most recent explicit (identical) mention, the part of speech of each word's most recent implicit (non-identical) mention, and the number of words elapsed between each word and its most recent previous mention.

4.4. Word Embedding Features

Previous work has shown that word embeddings can be useful for prosody prediction [8]. However, when training a random forest classifier like the one used for our models, we could not use word embeddings directly, as the individual dimensions are not useful unless used within a neural model. Therefore, we generated word and sentence embeddings, obtaining the latter by averaging the word embeddings of each word in a sentence. Then, to integrate these into the model, each word embedding was assigned to one of five clusters and each sentence embedding to one of twenty, based on the k-means clustering algorithm. We included the cluster IDs for the current word and sentence as features, as well as cluster IDs for neighboring words and sentences, using a context window size of three on each side.

We explored a large number of potential word embeddings, both pre-trained embeddings and embeddings trained on BURNC, in order to determine which performed best in this task. For phrase boundary detection test, we found that we achieved the best results from training word2vec directly on BURNC [22]. For pitch accent detection, we found that the best results were achieved when we used a set of pre-trained GLoVe embeddings that were modified to be gender-neutral [23]. These embeddings were probably helpful because any gender bias present in the data would be unlikely to have a bearing on prosody; therefore, these embeddings effectively removed irrelevant semantic content.

4.5. Speciteller

The last feature we included was the Speciteller score [24]. Speciteller is a tool that determines the level of specificity and detail present in a sentence. It assigns a rating ranging from 0 (most general) to 1 (most detailed) based on the words present in a sentence, with sentences containing many pronouns and vague terms like *people* having much lower scores than sentences containing specific terms or proper names. For example, the sentence *"Estimates vary widely on how much money could be saved"* was assigned a score of 0.0186, while the more specific sentence *"Quincy based Arbella Mutual Liability can now take over American Mutual's forty thousand car and home owner's policies"* was assigned a score of 0.872. We calculated a specificity score for each sentence in the corpus, and the same value was assigned to each word of the sentence. We found that Speciteller scores improved performance for both phrase boundary and pitch accent detection.

5. TTS Pipeline

For the actual speech synthesis steps of our pipeline, we used Merlin, an open source neural-net-based TTS system [25]. Merlin's pipeline included three major steps. First, the text was converted into HTS-style label files. Next, DNN-based duration and acoustic models were used to predict acoustic features from these label files. Finally, a vocoder (the open source WORLD vocoder, in this case) was used to convert from acoustic features to speech. For our work, we modified only the front end of this pipeline, keeping the duration model, acoustic model, and vocoding steps as is. A diagram of our modified pipeline is shown above in figure 1.

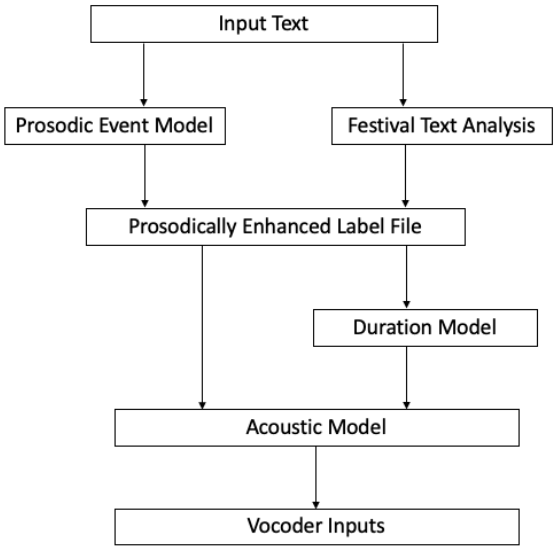

Figure 1: *A diagram of our prosodically enhanced Merlin TTS pipeline*

The HTS-style label files contain one line per phone, with a large number of features attached to each phoneme, including the identity of surrounding phones, whether the phone is in a stressed syllable or not, the phone's position in the current syllable and word, and many others. While these files do contain features relating to the phone's position in the phrase, in practice, we found that altering these features makes very little difference in the synthesized audio.

Instead, we added two extra features to each phone, one corresponding to phrase boundaries and the other corresponding to pitch accents. Since there are notable prosodic differences between 4-level breaks at the end of sentences and other 4-level breaks, the phrase boundary feature could take on one of three values: one for phones in words followed by a sentence break, one for phones in words followed by a mid-sentence 4-level break, and one for phones in words not followed by a 4-level break. For pitch accents, our feature was a simple binary feature, indicating whether the current phone belonged to a pitch accented word or not.

In order to add these features to each line of the label file, we ran our prosodic event prediction model on both our training and test data. For consistency's sake, since our training data for this stage included BURNC utterances without gold standard ToBI labels, we used only predicted labels for this step. However, since some utterances were in the training set in both steps, it is likely that the predicted labels were very similar to the gold standard labels.

We next ran the utterances through the normal process for creating HTS labels, using the festival software, with the notable change that we removed the prosody module. We removed this module because the festival prosody prediction uses its own, less accurate, phrasing model, which often inserts pauses

in infelicitous places in the utterance. Finally, we added the new break and accent features to each line, updated the Merlin questions file to reflect the new features, and trained Merlin using the default DNN hyper-parameters for the acoustic and duration models.

6. Results

6.1. Objective Metrics

Table 1: *MCD and RMSE of F0 of synthesized from our pipeline compared to a baseline.*

Model	MCD (dB)	RMSE (Hz)
Our pipeline	5.014	44.586
Baseline	5.053	45.016

We ran our newly developed TTS pipeline on BURNC, training on all utterances from female speakers, except for 121 utterances from speaker f3, which were held out as a test set. In order to test the whether our pipeline improved output, we tested it against a baseline, which used the standard Merlin pipeline trained on the same data. We then computed the mel-cepstral distortion (MCD) and the F0 root mean square error (RMSE) between our synthesized utterances and the original utterance in BURNC. These statistics are presented in Table 1, which demonstrates that our model performs better on both metrics. This indicates that our pipeline does in fact produce speech that is closer to natural human speech.

6.2. Listening Test

In order to test whether our model was useful for news data from other corpora, including lengthier utterances, we also tested our model on sentences from recent news data and crowd-sourced these for human labeling. Specifically, using the same model as above, we synthesized 20 short paragraphs (ranging in length from 1 to 4 sentences) from news stories on the National Public Radio (NPR) show *Morning Edition*, with breaks and accents predicted using the model presented above.

These twenty utterances were then presented to Amazon Mechanical Turk workers, who were provided with two versions of the utterance, one synthesized using this pipeline and the other synthesized using the baseline standard Merlin pipeline. The Turkers were then asked to select which of the two they found more natural. This was a forced choice task; there is no "no opinion" option. The utterances were ordered so that in a random half of the questions, the users were presented with the baseline first, and in the other half, they were presented with the prosodically trained utterance.

Five workers provided judgments on each of these 20 utterances, for a total of 100 judgments. Of these judgments, 80% rated the utterances with our new, additional prosodic features better than the baseline. These judgments demonstrate that the new TTS pipeline we have developed, using the features described above, is significantly preferred, with a p-value of $1.97 * 10^{-9}$.

7. Discussion and Future Research

In this paper, we have presented a pipeline that can produce prosodically appropriate synthesized speech for novel utterances, many of which are quite lengthy. Additionally, since this pipeline predicts prosodic features strictly based on text-based features that can easily be extracted using NLP tools, it is very likely this pipeline can be used on other corpora that have no gold standard ToBI labels.

However, while our voice is significantly better than the baseline, it still suffers from some noticeable voice quality issues. These issues are probably due to the small size of the BURNC corpus, as well as the limitations of Merlin, which uses a somewhat outdated TTS pipeline. In our future work, we plan to create a similar pipeline using an end-to-end system, since such end-to-end systems have been shown to produce higher quality speech for TTS. Additionally, since our TTS pipeline does not rely on the training corpus having any manual ToBI labels, we will be able to work with a much larger set of training data.

While we have shown that this pipeline works well for replicating the prosody of news data, it is unclear if it will work as well on other prosodically challenging domains, such as conversational speech. While our previous work has shown that our prosodic event prediction model does not generalize well to the conversational domain, there are several ToBI-labeled conversational corpora, such as the Switchboard Corpus and the Columbia Games Corpus, which will provide a strong starting point for work on this problem. In our future work, we will examine this problem further, using various conversational corpora as training data. These include corpora collected in our lab, including the Games and Switchboard corpora as well as the Columbia Cross-Cultural Deception Corpus (CxD).

Additionally, while our approach notably improves prosody, because we are only modeling phrase boundaries and pitch accents, there are aspects of prosody that it cannot properly synthesize. This problem is compounded by the fact that our model cannot currently distinguish between boundary tones or between high and low accents. One possible way to overcome some of these shortcomings, particularly when working with end-to-end systems, is to use our low-level approach to prosody alongside a sentence-level approach, such as adding style embeddings to the system. Since Tyagi et al. showed that conditioning style embeddings on linguistic features can improve prosody [15], this could also be a way to incorporate features like dialogue act tags, which have a large impact on prosody but a small impact on the location of prosodic events. In the future, we plan to perform experiments where we incorporate models to capture both word-level and sentence-level prosody.

8. References

[1] J. Hirschberg and P. Prieto, "Training intonational phrasing rules automatically for English and Spanish text-to-speech," *Speech Communication*, vol. 18, no. 3, pp. 281–290, 1996.

[2] K. Ross and M. Ostendorf, "Prediction of abstract prosodic labels for speech synthesis," *Computer Speech & Language*, vol. 10, no. 3, pp. 155–185, 1996.

[3] J. Hirschberg, "Pitch accent in context predicting intonational prominence from text," *Artificial Intelligence*, vol. 63, no. 1-2, pp. 305–340, 1993.

[4] J. Hirschberg and O. Rambow, "Learning prosodic features using a tree representation," in *Seventh European Conference on Speech Communication and Technology*, 2001.

[5] P. Koehn, S. Abney, J. Hirschberg, and M. Collins, "Improving intonational phrasing with syntactic information," in *2000 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 3. IEEE, 2000, pp. 1289–1290.

[6] N. Obin and P. Lanchantin, "Symbolic modeling of prosody: from linguistics to statistics," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 3, pp. 588–599, 2015.

[7] T. Mishra, Y.-j. Kim, and S. Bangalore, "Intonational phrase break prediction for text-to-speech synthesis using dependency relations," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 4919–4923.

[8] A. Rendel, R. Fernandez, R. Hoory, and B. Ramabhadran, "Using continuous lexical embeddings to improve symbolic-prosody prediction in a text-to-speech front-end," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 5655–5659.

[9] Z. Malisz, H. Berthelsen, J. Beskow, and J. Gustafson, "Controlling prominence realisation in parametric dnn-based speech synthesis," in *Proceedings of Interspeech 2017*, 2017, pp. 1079–1083.

[10] T. Fujimoto, K. Hashimoto, K. Oura, Y. Nankaku, and K. Tokuda, "Impacts of input linguistic feature representation on japanese end-to-end speech synthesis," in *10th ISCA Speech Synthesis Workshop. ISCA, Vienna, Austria*, 2019.

[11] H. Guo, F. K. Soong, L. He, and L. Xie, "Exploiting syntactic features in a parsed tree to improve end-to-end tts," *arXiv preprint arXiv:1904.04764*, 2019.

[12] P. Wang, Y. Qian, F. K. Soong, L. He, and H. Zhao, "Word embedding for recurrent neural network based tts synthesis," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 4879–4883.

[13] Y. Xiao, L. He, H. Ming, and F. K. Soong, "Improving prosody with linguistic and bert derived features in multi-speaker based mandarin chinese neural tts," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6704–6708.

[14] Y. Wang, D. Stanton, Y. Zhang, R.-S. Ryan, E. Battenberg, J. Shor, Y. Xiao, Y. Jia, F. Ren, and R. A. Saurous, "Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis," in *Proceedings of the 35th International Conference on Machine Learning*, 2018, pp. 5180–5189.

[15] S. Tyagi, M. Nicolis, J. Rohnke, T. Drugman, and J. Lorenzo-Trueba, "Dynamic Prosody Generation for Speech Synthesis Using Linguistics-Driven Acoustic Embedding Selection," in *Proc. Interspeech 2020*, 2020, pp. 4407–4411.

[16] M. Ostendorf, P. J. Price, and S. Shattuck-Hufnagel, "The Boston University radio news corpus," *Linguistic Data Consortium*, pp. 1–19, 1995.

[17] K. Silverman, M. Beckman, J. Pitrelli, M. Ostendorf, C. Wightman, P. Price, J. Pierrehumbert, and J. Hirschberg, "ToBI: A standard for labeling english prosody," in *Proceedings of the International Conference Spoken Language Processing*, 1992, p. 867–870.

[18] R. Sloan, S. S. Akhtar, B. Li, R. Shrivastava, A. Gravano, and J. Hirschberg, "Prosody prediction from syntactic, lexical, and word embedding features," in *10th ISCA Speech Synthesis Workshop*, 2019.

[19] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky, "The Stanford CoreNLP natural language processing toolkit," in *Association for Computational Linguistics (ACL) System Demonstrations*, 2014, pp. 55–60.

[20] J. W. Pennebaker, R. L. Boyd, K. Jordan, and K. Blackburn, "The development and psychometric properties of LIWC2015," The University of Texas at Austin, Tech. Rep., 2015.

[21] J. Kasai, B. Frank, T. McCoy, O. Rambow, and A. Nasr, "Tag parsing with neural networks and vector representations of supertags," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 1712–1722.

[22] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[23] J. Zhao, Y. Zhou, Z. Li, W. Wang, and K.-W. C. Chang, "Learning gender-neutral word embeddings," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018.

[24] J. J. Li and A. Nenkova, "Fast and accurate prediction of sentence specificity." in *AAAI*, 2015, pp. 2281–2287.

[25] Z. Wu, O. Watts, and S. King, "Merlin: An open source neural network speech synthesis system." in *SSW*, 2016, pp. 202–207.