The IRISA Text-To-Speech System for the Blizzard Challenge 2018

Pierre Alain, Gwénolé Lecorvé,
Damien Lolive, Antoine Perquin

IRISA, University of Rennes 1 (ENSSAT), Lannion, France

pierre.alain@univ-rennes1.fr gwenole.lecorve@irisa.fr
antoine.perquin@irisa.fr damien.lolive@irisa.fr

Abstract

This paper describes the implementation of the IRISA unit selection-based TTS system for our participation to the Blizzard Challenge 2018. We describe the process followed to build the voice from given data and the architecture of our system. It uses a selection cost which integrates notably a bottleneck DNN-based embedding prediction. Unit selection is based on a Viterbi-based algorithm with preselection filters used to reduce the search space. The system achieves average results compared to others.

Index Terms: speech synthesis, unit selection

1. Introduction

In recent years, research in text-to-speech synthesis essentially focused on two major approaches. The first one is the parametric approach, for which HTS [1] and DNN-based systems [2] are now dominating the academic research. This method offers advanced control on the signal and produces very intelligible speech but with a low naturalness. The second approach, unit selection, is a refinement of concatenative synthesis [3, 4, 5, 6, 7, 8, 9]. Speech synthesized with this method features high naturalness and its signal quality is unmatched by other methods, as it basically concatenates speech actually produced by a human being.

The 2018 challenge, as in 2017, is to build an expressive voice using children's audiobooks in English. The main difficulty with audiobooks, and in particular for children, is the change of characters and especially the imitation of animals (*i.e.* roars) as well as other sounds that may occur. For instance, in the data provided, a bell ringing signal is given to tell the child that he/she has to turn the page. Considering the expressivity of the voice, the different sounds and characters we can find in such books, the main challenges are phone segmentation and expressivity control.

In this paper we present the unit-selection based IRISA system for the Blizzard Challenge 2018. Basically, as audiobooks for children contain very expressive speech, one need a mean to control the degree of expressivity selected units have. To do so, the system is based on a preselection filter to separate the acoustic unit space into narrative or non-narrative speech and on a beam-search algorithm to find the best unit sequence. The objective function minimized by the algorithm is composed of a target cost and a join cost. The join cost relies mainly on acoustic features to evaluate the level of spectral resemblance between two voice stimuli, *on* and *around* the position of concatenation. For instance, distances based on MFCC coefficients and especially F0 are used [10, 11].

In our contribution we introduce a phoneme embedding predicted using a feed-forward DNN with a bottleneck layer working as an acoustic model. . In that model, timing information has been postponed until after the bottleneck layer learned with the speaker's data. Predictions are then used in the target cost to rank units based on their prosodic properties.

The remainder of the paper is organized as follows. Section 2 describes the voice creation process from the given data. Section 3 details the TTS system and further details about the feed-forward DNN are given in section 3.3. Section 4 presents the evaluation and results.

2. General voice creation process

As in 2017, this year the challenge focuses on audiobook reading for children in English. The goal is then to build a voice based on approximately 6.4 hours of speech data provided as a set of wave files with the corresponding text. The recordings correspond to a set of 56 books targeted at children aged from 4 years old.

2.1. Data preparation and cleaning

The very first step has been to clean the text and make sure that it was corresponding to the speech uttered by the speaker. Moreover, all the quotation marks have been checked to insure an easy detection of boundaries between narrative and direct speech. Some parts corresponding to too expressive speech were discarded at this step to avoid later problems during synthesis. Despite of this, we still have preserved the major part of the expressive content. This work and the sentence level alignment has been done manually using Praat [12].

Finally, as the signals were provided using different formats, we have converted all the speech signals to standard WAV with a sampling frequency of 44.1kHz for further processing. F0 is extracted using the ESPS algorithm [13] while pitch marks are computed using our own algorithm.

2.2. Segmentation and feature extraction

To build the voice, we first phonetized the text thanks to the grapheme-to-phoneme converter (G2P) included in *eSpeak* [14]. Then the speech signal has been segmented at the phone level using HTK [15] and standard forced-alignment. The acoustic models used for segmentation are learned using the data provided for the challenge.

Additional information is extracted from the corpus, like POS tags and syllables. Moreover, a label is associated to each word indicating if it is part of direct speech or not. The label is obtained based on the quotation marks in the text. The main idea with this label is to separate normal speech from highly expressive speech, usually present in dialogs.

All this information is stored in a coherent manner using the ROOTS toolkit [16]. All conversions and interactions between the different tools are also managed with this toolkit as,

for instance, conversions from IPA (output of *eSpeak*) to the ARPABET phone set used in the synthesis engine.

3. The IRISA system

3.1. General architecture

The IRISA TTS system [17, 18], used for the experiments presented in this paper, relies on a unit selection approach with a beam-search algorithm. The optimization function is divided, as usually done, in two distinct parts; a target and a concatenation cost [4] as described below:

$$U^* = \underset{U}{argmin} \left(W_{tc} \sum_{n=1}^{card(U)} w_n C_t(u_n) + W_{cc} \sum_{n=2}^{card(U)} v_n C_c(u_{n-1}, u_n)\right) \quad (1)$$

where U^* is the best unit sequence according to the cost function and u_n the candidate unit trying to match the n^{th} target unit in the candidate sequence U. The search process is done using the beam-search algorithm using a beam of size 30. $C_t(u_n)$ is the target cost and $C_c(u_{n-1}, u_n)$ is the concatenation cost. W_{tc}, W_{cc}, w_n and v_n are weights for adjusting magnitude for the parameters. Sub-costs are weighted in order to compensate magnitudes of all sub-costs as in [19]. The problem of tuning these weights is complex and no consensus on the method has emerged yet. [20] is a good review of the most common methods. In our case, weights have been adjusted empirically.

3.2. Join cost

The concatenation cost $C_c(u, v)$ between units u and v is composed of MFCCs (excluding Δ and $\Delta\Delta$ coefficients), amplitude, F0 and duration euclidean distances, as below:

$$C_c(u,v) = C_{mfcc}(u,v) + C_{amp}(u,v) + C_{F0}(u,v) + C_{dur}(u,v),$$

where $C_{mfcc}(u, v)$, $C_{amp}(u, v)$, $C_{F0}(u, v)$, $C_{dur}(u, v)$ are the sub-costs, resp., for MFCC, amplitude, F0 and phone duration.

3.3. Target cost

The target cost is composed of a classic target cost and a cost between embeddings. The first part of the target cost can be assimilated to a linguistic cost and uses the following criteria: is the phoneme ending a breath group, a word or a sentence; is the phoneme first of word; is the phoneme is long, nasal, with low or high stress; is the syllable has a rising pitch contour or not. This linguistic cost is computed as a weighted sum of feature values differences between the candidate and target units.

The second part is computing as the euclidian distance between embeddings of candidates and predicted targets. The model used for prediction is a feed-forward DNN with a bottleneck layer, as shown on figure1, working as an acoustic model where timing information has been postponed until after the bottleneck layer [21]. The bottleneck scheme is symmetrically designed: 11 hidden layers of sizes 1024, 512, 256, 128, 64, 32, 64, 128, 256, 512, 1024.

For a given phoneme, the model predicts acoustic features $\mathbf{a}_i$ of the i-th frame based on the linguistic feature vector $\boldsymbol{\ell}$. Those linguistic features provide information about the phoneme, e.g., its identity and the one of its neighbors, its position in the syllable/word/utterance it belongs to, etc. The timing information is encoded as two numerical features: the phone duration d in seconds and the relative position t_i of the frame i inside the phone. This timing information is useful to take into account the dynamics of acoustic features when realizing a phone, while its position allows us to obtain a phone-level embedding e.

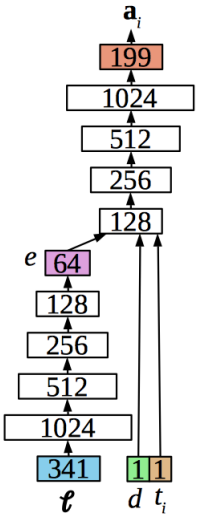

Figure 1: *DNN architecture.*

3.4. Dataset and experimental setup

Our models were trained on a corpus corresponding to the reading of an English audio-book by a professional English speaker resulting in approximately 6.4 hours of speech for a total of 1168 utterances. Speech is expressive (narration, acted dialogues), and sentences are heterogeneous in terms of length and complexity.

About 110 linguistic features are considered for each phone. Categorical attributes represent information about quinphones, syllables, articulatory features, and part of speech for the current, previous and following words. They are encoded as one-hot vectors. 34 other features are numerical, such as the position of the phone inside the word or the utterance. After encoding, the overall linguistic vector is of size 359. The linguistic features and the timing information were normalised to the range [0.01, 0.99]. Each linguistic feature was manually extracted, without automatic annotation.

The acoustic features, extracted using the WORLD vocoder [22], consist of a 60 dimensions Mel-Generalized Cepstral coefficients (MGC) vector, a 5 dimensions band-aperiodicity (BAP) vector and the fundamental frequency F_0. Those features have been extracted every 5 ms. The F_0 values have been linearly interpolated on unvoiced parts, a boolean attribute keeps track of whether the frame is voiced or not and the logarithm has been applied to F_0. Finally, the deltas and delta-deltas have been computed for MGC, BAP and F_0. In total, the acoustic vector is of size 199. The acoustic features have been centered and normalized to unit variance.

The implementation has been done using Keras with TensorFlow. Training has been done on a GTX 1080 Ti, over 250 epochs using RMSPROP with the mean square error as a loss

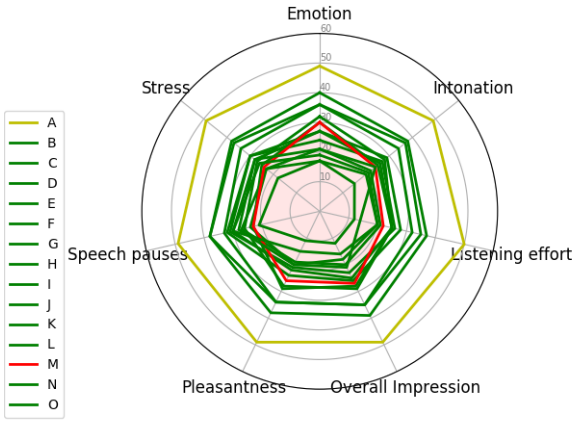

Figure 2: *Mean Opinion Scores for the different criteria and the systems evaluated. "Natural" system is shown in yellow and IRISA system in red while other participants are in green.*

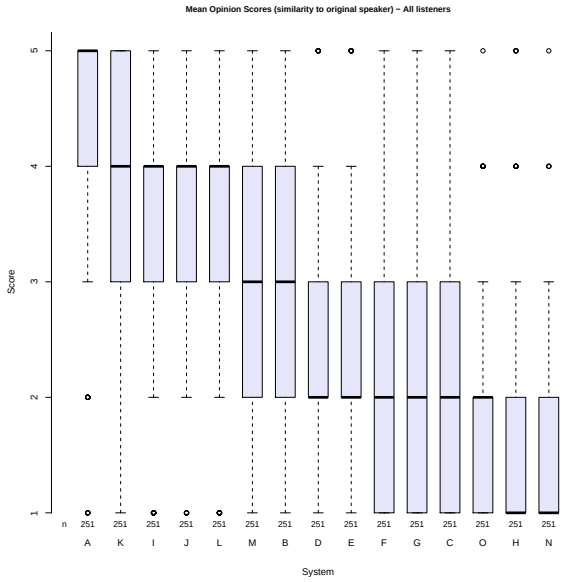

Figure 3: *Mean Opinion Scores, similarity to the original speaker evaluation, all listeners.*

function. The model weights with the best performance on the validation set have been saved. Those models have been trained using the true duration values.

4. Evaluation and results

The evaluation assessed a number of criteria (overall impression, speech pauses, intonation, stress, emotion, pleasantness and listening effort) for book paragraphs as well as similarity to the original speaker, naturalness and intelligibility. The evaluation has been conducted for different groups of listeners: paid listeners, speech experts, and volunteers. In this section, we only give results including all participants. In every figure, results for all 15 systems, including natural speech, are given. Among the systems, we have the following : system A is natural speech, System B is the Festival benchmark, (standard unit-selection voice built using the same method as used in the CSTR entry to Blizzard 2007), System C is the HMM benchmark built using the HTS toolkit and System D is the first DNN benchmark built using the HTS toolkit, System E is the second DNN benchmark built using the HTS toolkit, which employs trajectory training. System M is the system presented by IRISA.

4.1. Evaluation with paragraphs

Overall results are shown on figure 2 taking into account all listeners. For each criterion, our system achieves average results. These average results are likely to be explained by inconsistencies in the prosody and stress placement.

4.2. Similarity to original speaker and naturalness

The similarity of the speech produced, as shown on figure 3, is among the average systems with a mean score and a median value of 3.0. Similarly, naturalness is also quite good as shown on figure 4 with an average of 3.0 and a median value of 3.0. For naturalness, our system is comparable to the baseline festival system. Despite of that, those results are far from the best systems. Sometimes, the system performs very well but on average it makes many errors penalizing the similarity and the naturalness criteria. Main errors observed are artefacts due to concatenation.

4.3. Intelligibility

Concerning intelligibility, our system is comparable to other systems with an average word error rate of 44%. Detailed results are given on figure 5. Compared to last year, the corrections we made have improved the intelligibility, even if our system is not performing well on other criteria.

5. Discussion

Despite of the improvements we added to our system, the results are not satisfying. After inspecting the system configuration, it appears that some elements can be corrected quite easily and seem to have a large impact on the synthesis quality.

First, we now use the IPA alphabet from the start to the end of the sentence processing and thus we remove the different approximations from and to the ARPABET alphabet. To improve stress management, we have recently implemented a mechanism to relax stress constraints in case not enough units are present in the right context (first consider the use of the secondary stressed phoneme if we have no primary stressed candidates, then consider the use of a not stressed phoneme). Now the stress placement seems to be improved, at least during informal listening tests.

Then we use the work described in [23] to adapt the phonetization provided by Espeak to have more coherent phoneme sequences between the voice corpus and the ouput of the phonetizer. Moreover, we have better adjusted parameters of the join cost and the balance between join and target costs. Those last changes have a large impact on the output quality, which seems to be better now[1].

Finally, other parameters as the size of the beam for the

[1] samples are available on `https://synpaflex.irisa.fr/blizzard-2018`

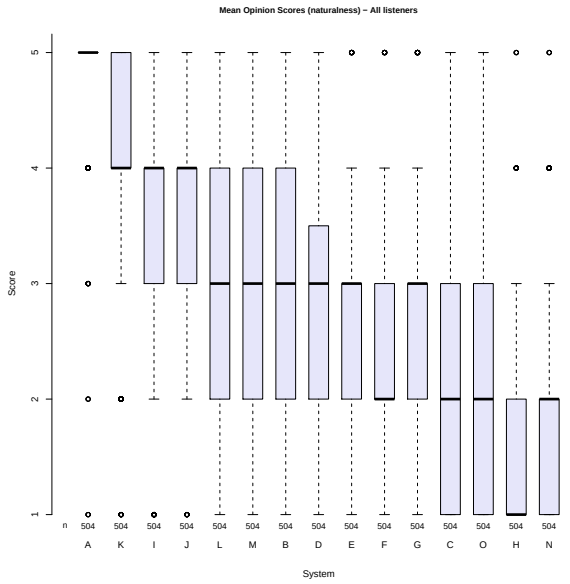

Figure 4: *Mean Opinion Scores, naturalness evaluation, all listeners.*

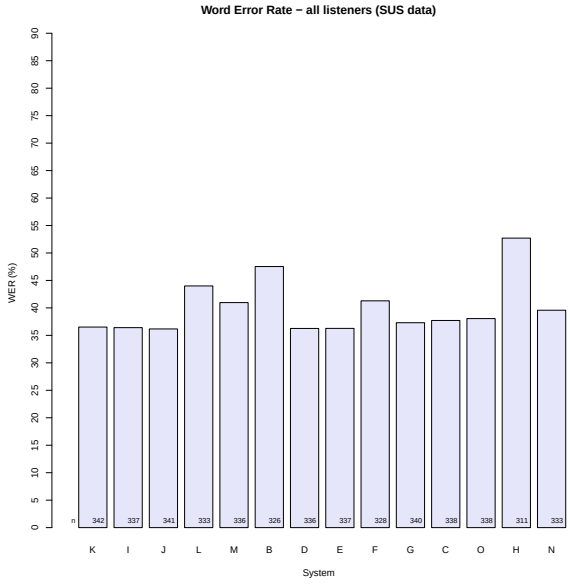

Figure 5: *Word Error Rates, intelligibility evaluation, all listeners.*

search, or the size of the candidates short list, are still difficult to tune. One important point is that those two parameters need to be chosen considering a trade-off between the number of constraints added during the unit selection and the variability of the corpus.

6. Conclusion

We described the unit-selection based IRISA system for the Blizzard challenge 2018. The unit selection method is based on a classic concatenation cost. In order to improve the system, we added specific feed-forward DNN to compute a target costs. Despite the changes we have made, our system obtained average results. With more time to prepare our input to the challenge we would have been able to do the necessary corrections that we have done after submission.

7. Acknowledgements

This study has been partially funded thanks to the ANR (French National Research Agency) project SynPaFlex ANR-15-CE23-0015.

8. References

[1] J. Yamagishi, Z. Ling, and S. King, "Robustness of HMM-based speech synthesis," in *Proc. of Interspeech*, 2008, pp. 2–5.

[2] H. Ze, A. Senior, and M. Schuster, "Statistical parametric speech synthesis using deep neural networks," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2013, pp. 7962–7966.

[3] Y. Sagisaka, "Speech synthesis by rule using an optimal selection of non-uniform synthesis units," in *Proc. of ICASSP*. IEEE, 1988, pp. 679–682.

[4] A. W. Black and P. Taylor, "CHATR: a generic speech synthesis system," in *Proc. of Coling*, vol. 2. Association for Computational Linguistics, 1994, pp. 983–986.

[5] A. Hunt and A. W. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," in *Proc. of ICASSP*, vol. 1. Ieee, 1996, pp. 373–376.

[6] P. Taylor, A. Black, and R. Caley, "The architecture of the Festival speech synthesis system," in *Proc. of the ESCA Workshop in Speech Synthesis*, 1998, pp. 147—-151.

[7] A. P. Breen and P. Jackson, "Non-uniform unit selection and the similarity metric within BTs Laureate TTS system," in *Proc. of the ESCA Workshop on Speech Synthesis*, 1998, pp. 373–376.

[8] R. A. Clark, K. Richmond, and S. King, "Multisyn: Open-domain unit selection for the Festival speech synthesis system," *Speech Communication*, vol. 49, no. 4, pp. 317–330, 2007.

[9] H. Patil, T. Patel, N. Shah, H. Sailor, R. Krishnan, G. Kasthuri, T. Nagarajan, L. Christina, N. Kumar, V. Raghavendra, S. Kishore, S. Prasanna, N. Adiga, S. Singh, K. Anand, P. Kumar, B. Singh, S. Binil Kumar, T. Bhadran, T. Sajini, A. Saha, T. Basu, K. Rao, N. Narendra, A. Sao, R. Kumar, P. Talukdar, P. Acharyaa, S. Chandra, S. Lata, and H. Murthy, "A syllable-based framework for unit selection synthesis in 13 indian languages," in *Proc. O-COCOSDA*, 2013, pp. pp.1–8.

[10] Y. Stylianou and A. Syrdal, "Perceptual and objective detection of discontinuities in concatenative speech synthesis," *Proc. of ICASSP*, vol. 2, pp. 837–840, 2001.

[11] D. Tihelka, J. Matoušek, and Z. Hanzlíček, "Modelling F0 Dynamics in Unit Selection Based Speech Synthesis," in *Proc. of TSD*, 2014, pp. 457–464.

[12] P. Boersma, "Praat, a system for doing phonetics by computer," *Glot international*, vol. 5, no. 9/10, pp. 341–345, 2002.

[13] D. Talkin, "A robust algorithm for pitch tracking (RAPT)," in *Speech coding and synthesis*, W. Kleijn and K. Paliwal, Eds. Elsevier Science, 1995, pp. 495–518.

[14] J. Duddington, "eSpeak text to speech," 2012.

[15] S. Young, G. Evermann, M. Gales, T. Hein, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev *et al.*, "The HTK book. for version 3.3 (april 2005)," 2013.

[16] J. Chevelu, G. Lecorvé, and D. Lolive, "ROOTS: a toolkit for easy, fast and consistent processing of large sequential annotated data collections," in *Proc. of LREC*, 2014, pp. 619–626.

[17] D. Guennec and D. Lolive, "Unit Selection Cost Function Exploration Using an A* based Text-to-Speech System," in *Proc. of TSD*, 2014, pp. 432–440.

[18] P. Alain, J. Chevelu, D. Guennec, G. Lecorvé, and D. Lolive, "The IRISA Text-To-Speech System for the Blizzard Challenge 2016," in *Blizzard Challenge 2016 workshop*, Cupertino, United States, Sep. 2016.

[19] C. Blouin, O. Rosec, P. Bagshaw, and C. D'Alessandro, "Concatenation cost calculation and optimisation for unit selection in TTS," in *IEEE Workshop on Speech Synthesis*, 2002, pp. 0–3.

[20] F. Alías, L. Formiga, and X. Llorá, "Efficient and reliable perceptual weight tuning for unit-selection text-to-speech synthesis based on active interactive genetic algorithms: A proof-of-concept," *Speech Communication*, vol. 53, no. 5, pp. 786–800, May 2011.

[21] A. Perquin, G. Lecorvé, D. Lolive, and L. Amsaleg, "Phone-level embeddings for unit selection speech synthesis," in *Proc. of the International Conference on Statistical Language and Speech Processing (SLSP)*, 2018.

[22] M. Morise, F. Yokomori, and K. Ozawa, "World: a vocoder-based high-quality speech synthesis system for real-time applications," *IEICE TRANSACTIONS on Information and Systems*, vol. 99, no. 7, pp. 1877–1884, 2016.

[23] M. Tahon, R. Qader, G. Lecorvé, and D. Lolive, "Improving tts with corpus-specific pronunciation adaptation," in *Proc. of Interspeech*, 2016.