EXEMPLAR-BASED PITCH CONTOUR GENERATION USING DOP FOR SYNTACTIC TREE DECOMPOSITION

Mohamed Abou-Zleikha, Peter Cahill, Julie Carson-Berndsen

CNGL, School of Computer Science and Informatics, University College Dublin, Dublin, Ireland

mohamed.abou-zleikha@ucdconnect.ie, peter.cahill@ucd.ie, julie.berndsen@ucd.ie

ABSTRACT

The generation of a pitch contour from linguistic information has long been recognised as a requirement for natural sounding speech synthesis. This paper investigates the use of an exemplar-based model for pitch contour generation. The main drawbacks of previous unit selection-based approaches for pitch contour generation is determining the size of the unit, and to guarantee that only prosodic and linguistically related units will be selected. The work presented in this paper overcomes these drawbacks by using only prosodic-syntactic correlated data, and a dynamic unit size model using data-oriented parsing. An AB comparison perceptual test showed 58% preference for the exemplar-based model, 25% for a HTS model, and 17% find both the same in terms of naturalness and pitch. In a MOS test, exemplar-based model achieved higher scores than that the HTS model achieved.

Index Terms: Speech synthesis, Intonation generation, Exemplar-based pitch generation,Prosody generation,syntactic-prosodic correlation

1. INTRODUCTION

State of the art speech synthesisers can achieve high mean opinion scores and low word error rates [1]. However, the difference between current synthesis technology and human speech is more clear-cut when the goal is to produce prosodically rich speech. Prosody modelling is an important factor in speech synthesis in which a pitch contour has a demonstrable role in the intelligibility and naturalness of synthesised speech [2]. It is one of the components (in addition to, but not limited to duration and energy) that contribute to the additional information in speech that is not in text. Several studies on speech synthesis have been undertaken in order to generate such information from text. Some of these studies tried to generate the contour using rule-based systems, others used statistical and exemplar-based methodologies [3, 4, 5, 6, 7, 8].

The work in this paper investigates using an exemplar-based approach to generate pitch contours. The proposed exemplar-based model aims to decompose the exemplars in an exemplar memory into meaningful sub-exemplars which are stored with rich information. When a new input is presented, it is decomposed into all possible combinations according to a composition function, and these combinations are evaluated in order to choose the best one. According to the chosen one, a target contour is generated.

2. PREVIOUS WORK

Several studies have investigated pitch generation from text. The motivation of many studies is to improve the naturalness in speech synthesis. Traditionally, syntactic trees, dependency trees and part of speech features (POS) in addition to the context features are used as potential text features.

Hidden Markov models can be used as statistical models for pitch generation [9] and also combined with decision trees for context representation [7]. Recurrent neural networks are also used [10, 11] for the purpose of pitch contour generation. Multi-layer perceptron and Elman network are also used for the same purpose as in [11]. Classification and regression trees was also a subject of study for pitch contour generation in [12]. The main problem with these methods is that they use average models to generate the contour. Unit selection based pitch generation was suggested with a good degree of success [4, 5, 6], where syllables are used as generation units. This approach showed promising results for pitch generation as it resulted in a natural contour. The problem was whether the generated pitch is a correct contour for the prosodic and linguistic context.

This work uses a unit selection model combined with an exemplar-based model as a pitch generation methodology, where dynamic unit size and linguistic and prosody related data are used. In the next section the pitch generation framework is explained in detail.

3. EXEMPLAR-BASED PITCH CONTOUR GENERATION

The presented exemplar-based generation method is inspired from the data-oriented parsing approach (DOP) which was suggested as an idea by Scha [13], but formalised and implemented by Bods [14]. DOP stores all previous language experiences. It operates by decomposing given exemplars into fragments and recomposing those pieces to analyse new utterances. Frequencies of structures are used to create probabilistic models for acquisition and processing. The exemplar-based generation model which is inspired from DOP consists of three steps:

1. Corpus Fragmentation
2. Solution Generator
3. Model Composition Operation

The first two steps are applied on the syntactic information; but the third one is applied on the pitch contour information in order to generate the pitch contour. Further explanation of each of these steps is discussed in following sections.

3.1. Data Fragmentation

Given a syntactic tree, the fragmentation process divides the tree into syntactically correct sub trees. The original tree can be generated by applying a substitution operation on each combination. The substitution operation identifies the nonterminal leaf node of one subtree

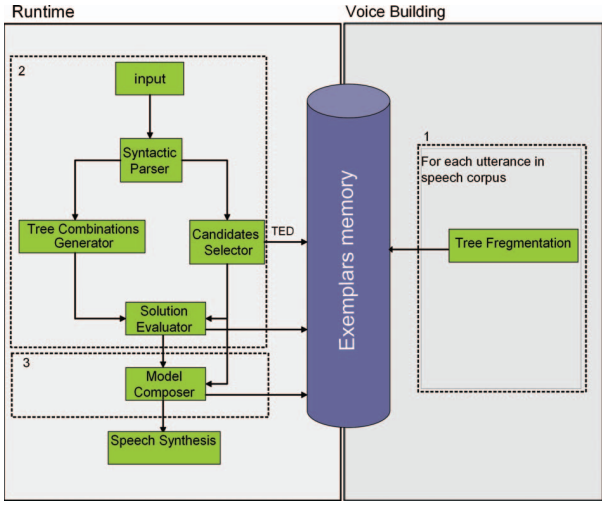

Fig. 1. *The general workflow for the generation process.*

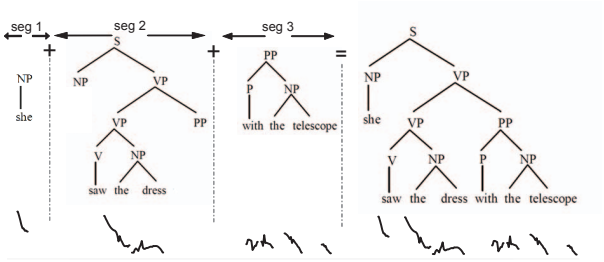

Fig. 2. *A possible fragmentation for a tree.*

with the root node of a second subtree in similar way to the original approach [14]. The tree can be formed using different combinations. The first stage of the process is to fragment the corpus and store the generated segments in the exemplar memory as illustrated as part 1 in Figure 1. The segments are stored in their phonetic representation. In case the generated segment is a one word segment (seg 1 in Figure 2), another structure is also generated which is the voiced non-voiced structure. This structure will be selected when the exact word is not found. Figure 2 illustrates a possible fragmentation for a tree. Each segment in the exemplar memory is associated with a pitch contour; this contour will be used in the model generation stage.

3.2. Solution Generator

The relationship between the syntactic information and pitch information was a question for previous research where a mapping between the syntactic information and pitch information is not one to one, but more a probabilistic relationship [15]. The paper shows that the mapping is determined by calculating the probability of a sample utterance chosen from the set which represents the closest $\alpha\%$ to an element on the text level being also an element of the set which represents the closest $\beta\%$ to that element on the prosody level (i.e. the conditional membership probability). By optimisation the value of this probability, these $\alpha\%$ from the data are determined and used for the solution evaluation.

The solution generator stage is applied when a new text input is presented and a pitch contour is required. The generation process uses the syntactic tree of the input text as the main feature for pitch generation. The generation process is applied by calculating:

1. the syntactic tree of the input text.
2. the possible combinations of the tree according to the fragmentation process.
3. the closest $\alpha\%$ of the corpus exemplars to the input tree according to syntactic tree distance function.

The combination that maximises a function f is selected as the optimal solution. This function is related to two factors, the frequency of the segment in the closest candidates in the corpus, and the number of syllables of the segments that do not exist amongst the closest candidates in the corpus. The function f is defined as

$$f(X = (x_1, .., x_n)) = \frac{\lambda * \prod_{j=1}^{m} P(x_j)}{(\mu * \sum_{k=1}^{l} C(x_k)) + 1} \quad (1)$$

where n is the number of segments in solution X, m is the total number of segments that exist in the corpus, l is the number of segments that do not exist in the corpus, $n = m + l$, $P(x_j)$ is the probability of segment x_j if it exists in the corpus, $C(x_k)$ is the number of syllables in the segment x_k when x_k does not exist in the corpus, λ and μ are the weights of the segment frequency and the number of missing syllable factors respectfully.

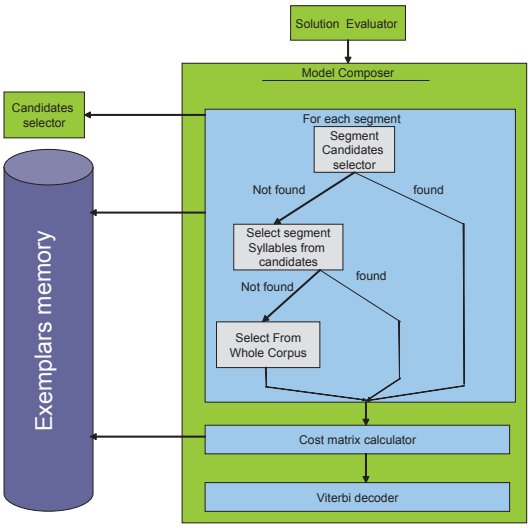

Fig. 3. *The process of pitch contour generation.*

This process is illustrated as part 2 in Figure 1.

3.3. Model Composer

As a result of the previous step, a solution for the input that consists of a set of segments is generated. The combinations of the pitch of these segments represents the target pitch contour. Each of these segments exists several times in the corpus, and the task now is to select the most suitable combinations in order to generate a consistent pitch contour. Some of these segments may not exist. For this purpose, the syllables that consist of the missing segments are selected instead. First they are selected from the closest candidates to the syntactic tree of the input text, but if not found the rest of corpus is considered. To select the optimal combination of the pitch segments

that generate the pitch contour, a unit selection approach is applied (which is illustrated as part 3 in Figure 1). The join cost is defined as the distance between the candidate pitch boundaries, and the target cost is defined as the distance between the context of the unit if the segment is syllables and 0 elsewhere. Figure 3 illustrates the process of the model composer. The optimal set of units is chosen and the pitch segments of these units are concatenated to generate the pitch contour.

4. EXPERIMENTS

To validate the proposed model, it has been integrated with a diphone unit selection speech synthesiser [1] and a set of perceptual tests has been conducted.

4.1. Data description

Blizzard Challenge 2011 corpus was used. The entire voice data consists of 15 hours and 6 minutes of audio. This data is split into 12095 utterances, of which 909 utterances were held-out of the voice as they contained words that are not in Unilex dictionary and therefore may have incorrect phonemes or syllables estimated. As a result, 677,830 diphones units were in the voice. 365,408 segments are generated as a result of the fragmentation process for the corpus. The Stanford parser [16] is used to generate the syntactic trees for both the corpus and the input data. The tree edit distance function (TED) (which represents the number of required transformation operations to transform the tree representation of one text to the tree representation of another) is used to select the closest α candidates from the corpus [17] to the input. Studying the corpus shows that the probability of finding the closest 10% utterances according to TED in the closest 30% utterances according to pitch similarity is 0.7. The closest 10% of utterances according to TED was used in the experiments.

4.2. Evaluation

26 utterances were used for the tests from 4 categories; *conversation*, *novel*, *news* and *reportorial*. Each utterance was synthesised using the proposed pitch generation method and the typical HTS f0 generation method (5 states multi-space probability distributions (MSD) HMM). Hidden semi Markov model (HSMM) is used for duration modelling, and no use for the generated MFCC models. Two tests have been performed in order to evaluate the performance of the system:

1. AB comparison test: the purpose of this test is to validate the generated pitch contour against the typical HTS generated pitch contour, according to the naturalness of the speech and the pitch model.
2. Mean opinion scores (speech naturalness).

The experiments consisted of 22 participants; each of them heard and evaluated 10 randomly selected synthesised utterances.

4.2.1. AB Comparison test

The AB test is performed to compare utterances synthesised using the proposed model and the HTS model. Each participant has to listen to ten pairs of utterances. Each pair consisted of two recordings with the same utterance generated by the two different methods under investigation. After listening to a pair of sentences, the participant has to answer questionnaire of the form:

1. Which one is more natural?
2. Which one has better pitch?

Figures 4 and 5 illustrate the results of the perceptual experiment to compare the speech that was synthesised using the exemplar-based model (EB) and using the HTS model (HTS) for each of the described questions.

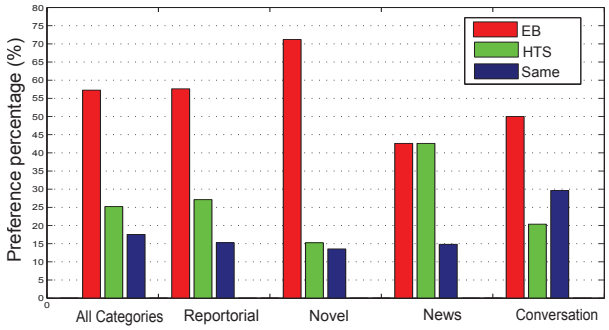

Fig. 4. *The percentage of votes for the EB and the HTS models with respect to the naturalness of the speech.*

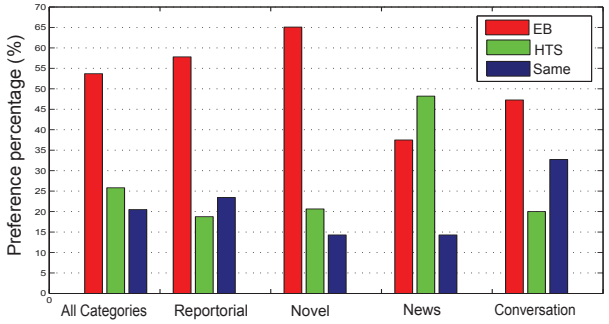

Fig. 5. *The percentage of votes for the EB and the HTS models with respect to the pitch.*

On the naturalness level, using all of the categories, 58% of the participants preferred the EB over HTS, while 25% preferred HTS, and 17% found that the two models have the same level of naturalness. The participants preferred the EB model for *reportorial*, *novel* and *conversation* categories; while participants found both models having the same naturalness level for *news*. The utterances from the *novel* category generated by EB model was the most preferred. On pitch level, similar results to the naturalness level have been found with a difference in the *news* category, where the it is found that the pitch that is generated using the HTS model is better than the ones generated by the EB model. This might be due to that less of personalised pitch contour is needed in such category and the average models in the statistical approach serves well in such case.

4.2.2. Naturalness test

Figure 6 illustrates the MOS for each category. EB achieved statistically significant better scores in *reportorial*, *novel* and *conversation* categories; while the both models achieved similar results in the *news* category. Using all data, EB achieved statistically significant better scores.

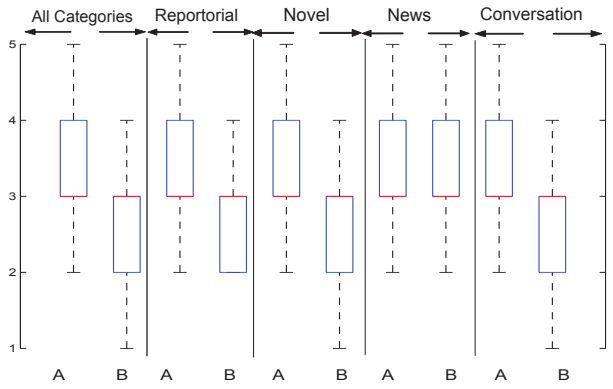

Fig. 6. *Naturalness scores (MOS) for all data and for each category for System A (Exemplar-based generation model) and B (HTS generation model).*

5. DISCUSSION AND CONCLUSION

This paper presents an exemplar-based model for pitch contour generation. The model consists of two stages, a data preparation stage where the syntactic tree is fragmented to its grammatically correct subtrees, and generation stage where for a new input text, a syntactic tree is extracted, and then the possible combinations of subtrees that generate the input tree are formed; these combinations are evaluated according to the frequency of their components occurrences in the corpus. The optimal combination is used to generate the new pitch contour by calculating the distance between the candidates of each segment and applying Viterbi decoding to extract the best combination between these candidates.

To validate this model, a set of experiments are performed. The first experiment consists of a comparison between the proposed model and MSD HMM pitch generation model. The comparison shows that EB is more natural, especially for the *novel* category. With respect to pitch preference, similar results of the naturalness level have recorded with a slight advantage for HTS model in the *news* category, where the participants found that the pitch generated using HTS is more suitable. The second test was a MOS test to investigate the naturalness of the synthesised speech as an independent model. EB has achieved better score in *reportorial*, *novel* and *conversation* categories; while the both models achieved similar results in *news*.

Future work includes investigating the effects of number of missed syllables on the model performance, conducting tests for the model on HMM-based speech synthesis and investigating the exemplar-based model for duration generation. Further improvement of the process of the contour generation stage is planned.

6. ACKNOWLEDGEMENTS

This research is supported by the Science Foundation Ireland (Grant 07/CE/I1142) as part of the Center for Next Generation Localisation (www.cngl.ie) at University College Dublin, Ireland. The opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of Science Foundation Ireland.

7. REFERENCES

[1] P. Cahill, U. Ogbureke, J. Cabral, E. Szekely, M. Abou-Zleikha, Z. Ahmed, and J. Carson-Berndsen, "Ucd blizzard challenge 2011 entry," *Proceedings of Blizzard Challenge Workshop*, 2011.

[2] W. Hess *et al.*, "Pitch determination of speech signals: algorithms and devices," 1983.

[3] J. Vroomen, R. Collier, and S. Mozziconacci, "Duration and intonation in emotional speech," in *Third European Conference on Speech Communication and Technology*, 1993.

[4] F. Malfrere, T. Dutoit, and P. Mertens, "Automatic prosody generation using suprasegmental unit selection," in *3rd ISCA Workshop on Speech Synthesis*, 1998.

[5] J. Meron, "Applying fallback to prosodic unit selection from a small imitation database," in *Seventh International Conference on Spoken Language Processing*, 2002.

[6] A. Raux and A. Black, "A unit selection approach to f0 modeling and its application to emphasis," in *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2003.

[7] H. Zen, K. Tokuda, and A. Black, "Statistical parametric speech synthesis," *Speech Communication*, 2009.

[8] P. Taylor, S. King, S. Isard, and H. Wright, "Intonation and dialog context as constraints for speech recognition," *Language and Speech*, 1998.

[9] T. Toda and K. Tokuda, "Speech parameter generation algorithm considering global variance for hmm-based speech synthesis," in *Ninth European Conference on Speech Communication and Technology*, 2005.

[10] S. Chen, S. Hwang, and Y. Wang, "An rnn-based prosodic information synthesizer for mandarin text-to-speech," *IEEE Transactions on Speech and Audio Processing*, 1998.

[11] A. Sakurai, K. Hirose, and N. Minematsu, "Data-driven generation of f0 contours using a superpositional model," *Speech Communication*, 2003.

[12] K. Hirose, K. Sato, and N. Minematsu, "Emotional speech synthesis with corpus-based generation of f0 contours using generation process model," in *Proc. International Conference on Speech Prosody, Nara*, 2004.

[13] R. Scha, "Taaltheorie en taaltechnologie; competence en performance," *Computertoepassingen in de Neerlandistiek, Almere: Landelijke Vereniging van Neerlandici (LVVN-jaarboek)*, 1990.

[14] R. Bod, "Exemplar-based syntax: How to get productivity from examples," *Linguistic review*, 2006.

[15] M. Abou-Zleikha and J. Carson-Berndsen, "Correlating text with prosody," in *12th Annual Conference of the International Speech Communication Association*, 2011.

[16] D. Klein and C. Manning, "Accurate unlexicalized parsing," in *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, 2003.

[17] E. Demaine, S. Mozes, B. Rossman, and O. Weimann, "An optimal decomposition algorithm for tree edit distance," *Automata, languages and programming*, 2007.