RECENT IMPROVEMENTS OF PROBABILITY BASED PROSODY MODELS FOR UNIT SELECTION IN CONCATENATIVE TEXT-TO-SPEECH

Wei Zhang, Liang Gu, Yuqing Gao

IBM T. J. Watson Research Center, Yorktown Heights, NY 10598
zhangwe@us.ibm.com

ABSTRACT

The work presented in this paper is subsequent to the paper "Probability Based Prosody Model for Unit Selection" which was published in ICASSP'2004. In the improved probability prosody model for corpus based concatenative Text-to-Speech (TTS), likelihood is replaced with posterior probability in the cost functions which conduct the following step, unit selection. Objective and subjective experiments show that posterior probability has obvious advantages over likelihood on robustness, flexibility and overall quality.

Index Terms— Posterior probability, prosody model, unit selection, Text-to-Speech (TTS)

1. INTRODUCTION

As we all know, Corpus-based concatenative TTS is still the most popular approach for speech synthesis although a lot of challenges in this direction need to be solved. Prosody models play very important roles in concatenative TTS. General speaking, the task of prosody models in concatenative TTS is to predict pitch, duration and energy values in some explicit or implicit forms. The predictions help to find the best matched candidate segments from the corpus. At the beginning, the approaches of prosody modeling for concatenative TTS are derived from the rule-based or other traditional TTS systems of that time [6, 7, 8], such as ToBI based pitch model and Fujisaki's command based pitch model. Those approaches have good flexibility and some modifications can be performed if the candidate doesn't exactly match the predictions. But it is difficult to keep natural and crisp voice quality. Later, in order to take the advantage of big corpora, statistics approaches are popularly adopted. There are two trends. One trend is to predict the explicit targets as the traditional models [3, 4]. Over-smooth is a big challenge to be handled in this trend. The other trend is to give soft prediction such as GMM and the selected candidates are expected to be best matched from probability point of view [1]. It is very difficult to assign weights for multiple prosody models and other non-prosody models. As the corpora built for TTS become bigger and bigger, and customers' expectations in terms of quality become higher and higher, some approaches try to keep away from complicated prosody models [9,10], but it substantially sacrifices the flexibility. Nowadays, we are still facing a lot of challenges on how to evaluate or tune a given prosody model because the criteria of "good" prosody is not quantitatively defined yet. It is also difficult to balance between flexibility and quality.

The paper is organized as follows. In Section 2, the disadvantages of likelihood function and how posterior probability makes up for those disadvantages are discussed. In Section 3, the implementation details about posterior probability based prosody model are presented. In Section 4, three experiments are done to verify these hypotheses. In Section 5, conclusions are given. Contributors are acknowledged in the last section.

2. LIKELIHOOD V.S. POSTERIOR PROBABILITY

2.1 Briefing of likelihood function based prosody models

In the approach presented in [1], six prosody models are built to give prosody predictions and then generate sub-costs for the following unit selection. The six models are prosody target model, prosody transition model, duration target model, duration transition model, energy target model and energy transition model. For each model, all the available data are used to train a context dependent decision tree and the clustered data in each leaf are abstracted as a Gaussian mixture. Usually, one to three Gaussians are generated in each Gaussian mixture.

In runtime, for each synthesis node, syllable or phone, each decision tree, T_i, is traversed separately according to the context of the node to get the corresponding Gaussian mixture M_i^l. l indicates the lth leaf under the decision tree T_i. For each available synthesis candidate x, the parameter sub-vector x_i, which is corresponding to the tree T_i, can be retrieved from the pre-built database. The minus log of the likelihood is defined as the cost of the candidate x to T_i under the given context, as formula (1)

$$C(x_i,\ M_i^l) = -\log P(x_i \mid M_i^l) \quad (1 \le i \le 6) \quad (1)$$

In the unit selection step, beam search are performed. Costs generated by target models and transition models are accumulated with simple weights as target cost and transition cost. The cost function is integrated with some other cost functions generated by acoustic or phonetic models to fulfill the search and get the best candidate path. Concatenation is performed at the end on the candidates in the best path to get the eventual synthesized speech output.

2.2 Disadvantages of likelihood function based prosody models

While the likelihood function based prosody model achieved very good results on several languages including Mandarin and English [1, 2], some serious problems are also found one by one in the past years.

The first problem is that the cost can be smaller than zero. As $P(x_i \mid M_i^l)$ is the probability density of model M_i^l at vector x, the value can be larger than 1. Accordingly, $C(x_i, M_i^l)$ can be smaller than zero. It is not reasonable as a "cost" definition. In unit selection, the concatenation costs between two candidate segments are usually set to be zero if they are also consecutive in the recording corpus. Smaller-than-zero sub-cost can introduce big confusions.

The second problem is that each Gaussian mixture is to optimize the output locally, not globally. Some leaves with smaller clustered data in the trees can bring big variances in cost functions. For well known reasons, some leaves have little data, and the distribution $P(x_i \mid M_i^l)$ is sometimes sharp. In the cost function, small variance among candidates can have substantially different cost values. But usually, these leaves are not important. It also happens in over-train situation. A cluster can be split to be two or more clusters when over-train happens. These clusters are regarded as independent ones and may dominate the search cost if the corresponding models are chosen.

The third problem is difficult to tune the weights for different models. The individual Gaussian mixtures have totally different distributions, and there are different dimensions for each model, and the trees can grow to different sizes without obvious overtrained or undertrained indications. It is difficult to find a good way to tune the weights. Rich language specific skills are required for the weight tuning, but the final result is always not the best.

2.3 Advances of Posterior Probability

Posterior probability is defined as $P(M_i^l \mid x_i)$. It can be calculated with formula (2). The cost, $C_{new}(x_i, M_i^l)$, is defined as minus log of $P(M_i^l \mid x_i)$ as formula (3).

$$P(M_i^l \mid x_i) = \frac{P(x_i \mid M_i^l) P(M_i^l)}{\sum_{j=1}^{N_i} P(x_i \mid M_i^j) P(M_i^j)} \quad (2)$$

$$\begin{aligned} C_{new}(x_i, M_i^l) &= -\log P(M_i^l \mid x_i) \\ &= -\log \frac{P(x_i \mid M_i^l) P(M_i^l)}{\sum_{j=1}^{N_i} P(x_i \mid M_i^j) P(M_i^j)} \end{aligned} \quad (3)$$

In (2) and (3), N_i is the Gaussian mixture number of the given decision tree T_i, as known as the leaf number of the tree T_i.

Theoretically, formula (3) can avoid or reduce the negative impacts of the three disadvantages in formula (1).

Obviously, in formula (2), both numerator and denominator are positive and the numerator is always smaller than denominator. It means $C_{new}(x_i, M_i^l) \ge 0$ under any condition, which meets the requirements of cost definition.

The cost function $C_{new}(x_i, M_i^l)$ is related to not only M_i^l but also all the Gaussian mixtures under the tree. When M_i^l is trained from a leaf with data sparse issue, $P(M_i^l)$ can reduce the weight of the cost. And $C_{new}(x_i, M_i^l)$ can be zero only if x's probability output in other models $M_i^{j \ne l}$ are zero. No doubt, posterior probability can achieve global optimization better than likelihood function. It can also reduce the negative impacts by over-train issues

Assume the distribution of the models under the decision tree is so extremely flat that all $P(x_i \mid M_i^{1 \le j \le N})$ are equal, then

$$C_{new}(x_i, M_i^l) = N_i \quad (4)$$

Under this assumption, $C_{new}(x_i, M_i^l)$ is correlative to the size of tree. It brings a possibility that leaves number of the decision tree can be used to adjust the weight of the sub-costs. It is reasonable. For some important features such as pitch, the tree always has more leaves. For some relatively not important features such as energy, the tree always has fewer leaves.

Some experiments are designed to verify the three hypotheses in the following sections.

3. IMPLEMENTATION

The new approach was implemented in the Iraqi Arabic TTS which was an important component of Multilingual Automatic Speech-to-speech Translator as known as MASTOR for English-to-Iraqi. The Iraqi-Arabic TTS was built with around 5-hour recordings and some more text scripts for linguistic analysis.

The system was combined with two parts. The first part performs language-specific processing. In this part, statistics approach based text vowelization[5], rule based syllable boundary determination, rule based stress assignment and rule based graphs to phone are performed. The output of this part is a labeled text stream. The second part performs prosody prediction which is the major focus of this paper, context-dependent acoustic prediction, and time-domain concatenation. All the components in the second part are language-independent which are supposed to be good for other languages.

As mentioned before, prosody modeling is the focus of this paper. In this component, three target models and three transition models are built for pitch, duration and energy separately. Pitch models are based on syllable unit. Duration and energy models are based on phone unit.

As usual, pitch models are the most complicated. Fig.1 shows the schema of parameters for pitch trees.

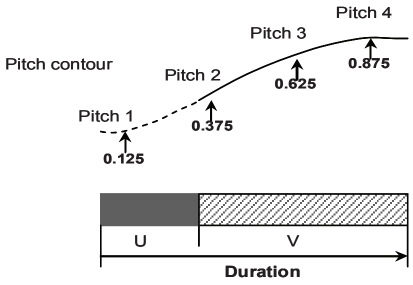

Fig. 1 Parameters of pitch target model

For pitch target model, x is compound of 5 parameters of the syllable unit. Four log pitch values at position of 0.125, 0.375, 0.625 and 0.875 separately, and one duration parameters. The duration parameter is introduced to distinguish the samples with similar pitch values but different duration values. Some syllables have unvoiced segments at the beginning or end, an interpolation approach is applied to make up for the pitch contour. For pitch transition model, x is compound of 2 parameters, difference between 1st pitch value of current syllable and 4th pitch value of previous syllable, and difference between 2nd pitch value of current syllable and 3rd pitch value of previous syllable. x of Duration target model has 1 dimension , which is the duration of the phone unit. The duration difference between current phone and previous phone is used for duration transition model. For energy target model, two parameters are used. Which are the STEPS(Short Time Energy Per Sample) values of the first half and the second half of the phone unit. The difference of the first STEPS value of current phone and the second STEP value of the previous phone is used for energy transition model.

Table 1. gives the leaves numbers of the corresponding trees.

Tree Type	Leaf Number
Pitch target prediction	421
Pitch transition prediction	190
Duration target prediction	112
Duration transition prediction	70
Energy target prediction	104
Energy transition prediction	110

Table 1: Leaf numbers of 6 decision trees

The sizes of the models are comparable with what we built for English and Chinese. Because few Iraqi-Arabic linguistic information can be collected, no special tuning on the sizes. For each leaf in the models, 1-3 Gaussian are generated as a mixture.

In target models, $\sum_{j=1}^{N_i} P(x_i \mid M_i^j) P(M_i^j)$ of each candidate segment can be pre-calculated. But the transition models are not so straightforward as the target models, because in transition models, x_i is generated dynamically according two consecutive segments in the searching path. In order to achieve same level CPU efficiency, vector quantization (VQ) is applied to generate a grid and the value of $\sum_{j=1}^{N_i} P(x_i \mid M_i^j) P(M_i^j)$ for each node in the grid is pre-calculated and saved. In runtime, for each x in transition models, the value of the closest node in the grid can be retrieved. For both target and transition models, $P(M_i^j)$ can be pre-calculated. So the CPU efficiency is kept in the same level as likelihood function based approach.

4. EXPERIMENTS

In the first experiment, 20 sentences were randomly selected from a test set which were excluded from the corpus of TTS voice. The 20 sentences were synthesized with two functions respectively, likelihood and posterior probability. In the final best path for each sentence, the ratios of each sub-cost to final cost were counted. Then the means and standard variances were calculated. The results were listed in table 2.

	Likelihood	Posterior
Pitch Target	0.378 ±0.033	0.181 ±0.025
Pitch Transition	0.216 ±0.043	0.188 ±0.031
Duration Target	0.024 ±0.010	0.171 ±0.008
Duration Transition	0.042 ±0.011	0.145 ±0.011
Energy Target	0.176 ±0.019	0.168 ±0.015
Energy Transition	0.165 ±0.015	0.148 ±0.009

Table 2. Ratio/Variance of sub-cost to final-cost

The results shows the ratios with likelihood function are very variable, but the ones with posterior functions are comparable. The standard variances of both results for each sub-cost are almost equal. It proves that the weight of sub-cost isn't correlated with parameters or sizes of the GMM models, which means the posterior probability function has advantages to reduce the negative impacts of the over-train issues and weight adjustment issues. The ratios of sub-costs are positively correlated with the sizes of decision tree although it is not such obvious linear correlation as we discussed before. It is understandable because that assumption we discussed is an extreme scenario which doesn't exist in the real datasets.

The second experiment was MOS evaluation with scale 1-5 as usual. Only three listeners are available, and 100 sentences were used as stimuli. The result was shown in Fig.2. The MOS was improved by 0.2.

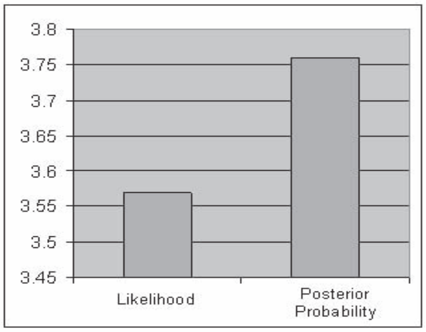

Fig. 2 MOS Evaluation

5. CONCLUSIONS

This paper presents the recent result of posterior probability function based prosody modeling approach which is improved from likelihood function based prosody modeling approach. The disadvantages in the old approach and how the new approach solves the issues are analyzed. The experiment results show the new approach has great advantages on robustness, flexibility and overall quality.

6. ACKNOWLEDGEMENTS

Thanks to Lazkin Tahir and Timothy Mathes for corpus recording, data processing and MOS evaluation. Thanks to Dr. Xiaodong Cui and Dr. Jia Cui for great support on speech recognition related tasks including corpus alignment and acoustic model building.

7. REFERENCES

[1] X.J. Ma, W. Zhang, W.B. Zhu, Q. Shi and L. Jin, "Probability Based Prosody Model for Unit Selection", ICASSP2004, Montreal, Canada.

[2] R. Fernandez, W. Zhang, E. Eide, R. Bakis, W. Hamza, Y. Liu, M. Picheny, J. Pitrelli, Y. Qin, Z. Shuang, L. Shen, "Toward Multiple-Language TTS: Experiments in English and Mandarin", Interspeech 2005, September 2005, Lisbon, Portugal.

[3] R. Donovan, E. Eide, "The IBM Trainable Speech Synthesis System", ICSLP'98, Sydney, Australia.

[4] E. Eide, A. Aaron, R. Bakis, P. Cohen, R. Donovan, W. Hamza, T. Mathes, M. Picheny, M. Polkosky, M. Smith, M. Viswanathan, "Recent Improvements to the IBM Trainable Speech Synthesis System", ICASSP 2003, Hong Kong, China.

[5] L. Gu, W. Zhang, L. Tahir, Y.Q. Gao, "Statistical Vowelization of Arabic Text for Speech Synthesis in Speech-to-Speech Translation Systems" Interspeech 2007, August 2007, Antwerp, Belgium

[6] A. Syrdal, C. Wightman, A. Conkie. Y. Stylianou. M. Beutnagel, J. Schroeter, V. Strom, K.S. Lee, M. Makashay, "Corpus-based Techniques in the AT&T NextGen Synthesis System", ICSLP 2000, Beijing, China.

[7] G. P. Kochanski and C. Shih, "Stem-ML: Language independent prosody description", ICSLP 2000, Beijing, China.

[8] J. P. H. van Santen, "Assignment of segmental duration in text-to-speech synthesis," Computer Speech and Language, 1994.

[9] M. Chu, H. Peng and E. Chang, "A concatenative Mandarin TTS system without prosody model and prosody modification", Proceedings of 4th ISCA workshop on speech synthesis, Scotland, 2001.

[10] W. Hamza and J. Petrelli, "Combining the Flexibility of Speech Synthesis with the Naturalness of Pre-Recorded Audio: A Comparison of Two Approaches to Phrase-Splicing TTS", Interspeech 2005, Lisbon, Portugal.