Learning Continuous Representation of Text for Phone Duration Modeling in Statistical Parametric Speech Synthesis

Sai Krishna Rallabandi, Sai Sirisha Rallabandi, Padmini Bandi and Suryakanth V Gangashetty

International Institute of Information Technology- Hyderabad, India

saikrishna.r@research.iiit.ac.in, siri.gene@gmail.com, munnipadmini@gmail.com, svg@iiit.ac.in

Abstract

In this paper, we investigate the usage of a continuous representation based approach of the feature vector derived from input text to predict the phone durations in a Text to Speech(TTS) system. We pose the problem of predicting the duration as a data driven statistical transformation from the input text onto the feature space. First we present a method to map both the categorical and numeric features that are typically used into a continuous numeric representation and then model it as a form of Matrix Factorization to improve the representation. The proposed system is evaluated based on Root Mean Squared Error(RMSE) as the objective measure and Mean Opinion Score(MOS) as the subjective measure. We find that the system performs on par with the state of the art duration modeling systems both subjectively and objectively.

Index Terms: Text to Speech Synthesis, Duration Modeling, Artificial Neural Networks,Vector Space Model

1. Introduction

Duration is an important prosodic feature which contributes not only to the structure but also to the perceived meaning of an utterance. Its influence on the intelligibility and the naturalness of the synthesized speech has been extensively studied [1]. In a Text-to-Speech system(TTS), segment duration is predicted by a model which is trained on a database of pre-selected units with known durations. The predicted durations are then imposed on the selected units via signal processing or used to compute a duration component of the target cost in concatenative speech synthesis systems [2]. In statistical parametric synthesis sytems or expressive/emotional speech synthesis, explicit prediction of phone durations are shown to improve the performance [3]. Duration modeling also contributes implicitly during the calculation of the synthesized fundamental frequency contour [4]. Time scale prosodic modification algorithms [5] utilize the predicted (target) durations to calculate the time scale modification factors for the selected speech units. It is thus clear that duration modeling contributes to the most of the components that are usually utilized during the process of converting text to speech. Hence, accurate and robust duration modeling is one of the fundamental and interesting tasks in developing TTS systems.

In this section, we briefly describe the issues in the modeling of durations and some of the previous approaches towards modeling and predicting the phone durations.

1.1. Issues in Duration Modeling

There are two major issues in duration modeling: Data Sparsity and Factor Confounding. Data sparsity refers to the fact that of the total observed feature vectors, many will be low in frequency. However, the joint probability mass of all these rare vectors taken together is sufficiently large [6] to mean that they cannot simply be neglected. Similarly, factor confounding refers to the condition that different factors occur with unequal frequencies in the training database. Due to this, there are significant variations in the duration of segment based on within-word position and stress. For instance, the durations of vowels turn out to be shorter in word-final syllables than in non-word-final syllables, if stressed and unstressed vowels are analysed together [7]. But, unstressed vowels are shorter than stressed vowels and word-final syllables are five times more likely to be unstressed than stressed.

A robust model for predicting segment duration must address all of these issues. It should generalise well in order to successfully predict the duration of segments with rare (or previously unseen) feature vectors.

1.2. Previous Approaches

Earlier approaches to model and predict the segment durations have been rule based [8] where each segment is assigned a duration by a set of rules. In [6], a Sum of Products based approach was proposed in which the durations of segments were predicted based on the factors influencing the same. Typically, duration modeling is done by combining a set of linguistic and paralinguistic features as a feature vector and then training a model for prediction. Non-parametric statistical modeling methods are attractive to predict the duration from text. Classification and Regression Trees (CART) models [9] are used in Festival synthesis system [10]. Artificial Neural Networks (ANN) [11], Bayesian Networks, Gradient Tree Bossting [12] and Multivariate Adaptive Regression Splines (MARS) modeling [13] have been successfully applied to predict duration from text [14] [15] in addition to fusion based techniques [16]. MARS, Bayesian Networks, Fusion based methods using Support Vector Regression and ANN models have comparable prediction accuracy and outperformed CART models.

However, all of the mentioned approaches require hand tuning of the features and also a significant amount of knowledge about the language of interest. The typical features used include the identities of the segments expressed as discrete binary values, the identities of the surrounding units, part of speech of the word in which the segment is present, the stress level on the unit, etc. Although, it would be appropriate to have manual finetuning of the feature vector based on the language of interest and also the application at hand, it would as well be ideal to have a framework where the feature vector representation is obtained in an unsupervised manner. The unsupervised method can also be extended to the languages where minimum or no linguistic

resources are available. Corpus based unsupervised approaches minimize the required human knowledge to describe a certain phenomena, provide reasonable solutions for difficult modeling problems, and have comparable results to rule based/fine tuned models.

In this paper, we propose to use continuous valued representations for the feature vector at the input which can be derived from text in an unsupervised fashion. The scope of the current work is limited to the investigation into the applicability of continous representation of the feature vector to predict durations from text. We pose these questions and try to address them before progressing to more complex systems: Firstly, are features taken from continuous representation equally as useful for the task as the complete set of hand tuned feature vectors? Secondly, how big a decrease in performance is caused by using mean durations for each unit instead of any prediction at all? Can we simplify the training process required to arrive at the continuous representation?

The outline of the paper is as follows: In Section 2 we describe the various blocks of our proposed front-end system, and then present our method to improve the continuous representations in section 3. We follow this up with a description of the objective and subjective evaluation of the system in Section 4, and finally conclude with a summary of future directions.

2. Proposed Framework

We first preprocess and tokenize input text, and then pass the result to a letter-to-sound (LTS) converter which converts the text into units. The units are then used to obtain the distributed representations in the feature space which capture the semantic as well as paralinguistic properties specific to the language. Finally, we enhance the feature representations based on the task at hand. The prediction from the text can then be collated into a feature file and passed onto the back end of the system for speech generation.In this section, we describe briefly the individual modules starting with the choice of the unit level.

2.1. Choice of the Unit

Durations have been modelled from various linguistic and acoustic units, from state level to phone, syllable level and word level. We have chosen phone as our basic unit over other units because of the following reasons:

- Phone is the most basic linguistic unit and hence, a framework with phone as the basic unit requires very minimal linguistic knowledge. In addition, even though the durations are predicted at higher linguistic units, they have to be usually scaled down to the phone level during synthesis [15].
- The prosodic structure of utterances is known to be reflected in the acoustic realization of the constituent phones [7].
- The choice of phone alleviates the data sparsity problem. The number of phones is roughly around 40 which is easy to model compared to the number of syllables or words.
- While building the co occurence statistics, all the units are covered if the unit is phone as opposed to selecting a suitable number of prominent units in case of words/syllables.

2.2. Letter to Sound Converter

There are many techniques in the literature[17][18][19] to predict a stream of phones given an input stream of letters including decision tree approaches such as CART [20] [21]. For the current work, we train a letter-to-sound predictor by encoding the CMU Pronounciation Dictionary for converting letters to phones.

2.3. Distributed Representation

The distributional analysis is conducted via vector space models (VSMs). The VSM [22] was originally applied to the characterisation of documents for purposes of Information Retrieval [23]. VSMs are applied to Speech synthesis from text in [24], where prediction models are built at various levels of analysis (letter, word and utterance) from unlabelled text. To build these models, co-occurrence statistics are gathered in the form of matrix to produce high-dimensional representations of the distributional behaviour of the chosen unit in the corpus. Appropriate lower dimensional representations are obtained by approximately factorising the matrix of raw co-occurrence counts by the application of Singular Value Decomposition(SVD). The distributional analysis places textual objects in a continuous-valued space, which is then partitioned by decision tree questions during the training of TTS system components such as acoustic models for synthesis or decision trees for pause prediction. In [24] VSM of letters was constructed by producing a matrix of counts of immediate left and right co-occurrences of each unit type, and from this matrix a 5-dimensional space was produced to characterise the units. We build on the same technique and try to improve the obtained representations based on the task at hand.

3. Improving Distributed Representations

The baseline word representations were built using the Latent Semantic Indexing approach mentioned in [24]. However, it was a very generic implementation and inorder to cope with the two shortcomings mentioned - data sparseness and factor confounding, we propose the following representations and try to investigate the ability of the same to predict the durations.

3.1. Representations similar to Word2Vec

Recently, there has been a lot of work supporting the representation of words as dense vectors, derived using various training methods inspired from neural-network language modeling [25][26]. We propose to derive the distributed representation at the phone level similar to the approach used in SkipGram Model [27]. Inspired by [28], we pose the task as a matrix factorization problem and intend to solve it using Symmetric Singular Value Decomposition instead of having to use Stochastic Gradient Descent as in [27]. This makes the training process simple and fast.

3.2. Implementation as Matrix Factorization

SkipGram NegativeSampling Model(SGNS)[27] embeds both words and their contexts into a low-dimensional space, resulting in word and context matrices W and C. The rows of matrix W are typically used in NLP tasks (such as computing word similarities) while C is ignored. It is nonetheless instructive to consider the product

$$W.(C^T) = M$$

This way, SGNS can be described as factorizing an implicit matrix M of dimensions $|V_W|$ $|V_C|$ [28]. Typically the obtained representations are of 50 or 100 Dimensions in word embeddings. Thus, SGNS is factorizing a matrix in which each row corresponds to a word, each column corresponds to a context and each cell contains a quantity reflecting the strength of association between that particular word-context pair. On a closer observation at the objective function, it is apparent that the matrix which is being factorized is indeed the Point Mutual Information Matrix between the word and the context. Adapting it to the duration modeling, the training part can be posed as a problem of obtaining and then factorizing a Point wise Mutual Information Matrix. In the current formulation, PMI(p, c) measures the association between a phone p and a context c by calculating the log of the ratio between their joint probability (the frequency in which they occur together) and their marginal probabilities (the frequency in which they occur independently). PMI can be estimated empirically by considering the actual number of observations in the corpus.

$$PMI(p,c) = log\frac{(n_p c) * (n_d)}{n_p * n_c} \quad (1)$$

.

where

- $n_p c$ is the frequency of occurence of the phone in the corpus
- n_d is the size of the corpus
- n_c is the frequency of the occurence of the context in the corpus
- $n_p c$ is the frequency of the occurence of the phone IN the context and appearing in the corpus

Therefore, PPI(p,c) can be calculated from the co-occurence statistics collected in an unsupervised fashion. However, the rows of PMI might contain many entries of word-context pairs (p, c) that were never observed in the corpus, for which PMI(p, c) = log(0) = - ∞ and hence, the matrix is ill defined in its direct sense. One method to alleviate the issue is to smooth the probabilities using a Dirichlet prior by adding a small fake count (δ) to the underlying counts matrix, rendering all phone-context pairs observed [29]. The resulting matrix will not contain any infinite values, but it remains dense. Also, this means that observed but bad (uncorrelated) word-context pairs have a negative matrix entry, while unobserved (hence worse) ones have 0 (or small Dirichlet prior) in their corresponding cell. A sparse and consistent alternative from the NLP literature is to use the positive PMI (PPMI) metric, in which all negative values are replaced by 0:

$$PPMI(p,c) = max(PMI(p,c), 0)$$

While the PMI matrix emerges from SGNS with k = 1, it was shown that different values of k can substantially improve the resulting embedding. With k $>$ 1, the association metric in the implicitly factorized matrix is $PMI(p,c)$ - $log(k)$.

$$SPPMIk(p,c) = max(PMI(p,c)logk, 0)$$

Thus, the relations between the phone and the context in which it appears are represented using Shifted positive point wise mutual information matrix formed using the co occurence statistics.

3.3. Dimensionality Reduction using SVD over k - Shifted PPMI for Task specific training

Truncated Singular Value Decomposition (SVD) can be used as an alternative matrix factorization method to SGNSs stochastic gradient training with L_2 loss. However, in the SVD-based factorization, the resulting phone and context matrices have very different properties [28]. In particular, the context matrix C^{SVD} is orthonormal while the phone matrix P^{SVD} is not. On the other hand, the factorization achieved by SGNSs training procedure is much more symmetric in the sense that neither W^{W2V} nor C^{W2V} is orthonormal, and no particular bias is given to either of the matrices in the training objective. We therefore propose achieving similar symmetry with the following factorization:

$W^{SVD^{\frac{1}{2}}} = U_d.\sqrt{\Sigma}$

4. Experiments

4.1. DATA

For the unsupervised VSM building, we made use of the available large quantity of unannotated data- which amounts to 1.2 million tokens. The data used for the task of learning the durations is taken from the audiobooks : Pride and Prejudice and EMMA by Jane Austen. For each audiobook dataset, the speech and the corresponding text were segmented using the INTERSLICE tool [30], and CLUSTERGEN [31] voices were built within the Festival[32] and Festvox [33] frameworks.

4.2. Systems Built - Benchmark Systems

4.2.1. System D_{DT}: (Discrete Representations of Phones used in a CART framework) and System D_{NN}: (Discrete Representations of Phones used in a Neural Network framework)

System D_{DT} is a regression tree (CART) model, where the predictee is the duration of the phone and the predictors are the features associated with that phone. In this case, the features are represented in a 1-out of k fashion. In order to include context, we concatenate the features of the previous two phones and the next two phones with the feature of the phone under question. System D_{NN} is a neural network trained as a discriminative classifier. The system is trained using backpropagation and outputs the duration given the input features of a phone. As in System D_{DT} , we associate the features of the previous two phones and the next two phones with the phone under question. We use the 200L 500N 50N 500N 1N as architecture for this system, where L represents linear activation and N represents tangential (tanh()) activation.

4.2.2. System MD: Mean Values of Phone Durations used instead of determining Durations

System MD is a look up table model, where the mean durations of phones are used. At every occurence of the phone in the text, the mean value of the duration of the phone is used. The objective behind building this system was to quantify the amount of loss that occurs(here in terms of RMSE) if the average durations of the phones are used without the need for any kind of prediction.

4.2.3. Complete Systems FC_{CART} and FC_{NN}

Both the systems use all the linguistic features derived from the Festival utterance structure. We have obtained the features

Table 1: Performance (in terms of the MOS out of 5 and RMSE in milliseconds) of the various systems on the duration prediction task

Audiobook	Measure	MD	D_{DT}	D_{NN}	VSM_{LSA}	VSM_{PPI}	FC_{CART}	FC_{NN}
Emma	RMSE	54.3	39.2	39.15	32.51	**20.44**	19.22	20.02
Pride and Prejudice	RMSE	53.7	39.6	40.12	31.43	**20.17**	19.04	19.66
Emma	MOS	2.5	3.4	3.2	3.6	**4.1**	4.2	4.1
Pride and Prejudice	MOS	2.3	3.6	3.4	3.9	**4**	4.3	4

using dumpfeats feature of the Festival system and used them for the predicitons. For training NNs, the punctuation feature was represented using 1-of-k coding, and the positional features were normalised to have zero mean and unit variance.

4.2.4. Systems Built - Experimental Systems

VSM_{LSA} and VSM_{PPI} use the distributional representations in the input feature space. We have used the context of previous and the next two phones and applied the procedure described in Section 3 to obtain the representations at the phonemic level. The dimensions of the input feature vector were hence 100 (20 dimensions for the phone and 20*4 for the context) For each corpus 10 percent of the data was a held out test set while the remaining 90 percent was used as a train set. The input dimensions were ZScore normalized and the output durations were normalized to be in the range of 0.01 to 0.99. An important issue here is how phones in unseen contexts are handled at test time. We follow a similar approach to that in [24]. We take a portion of the train set (we use 5 phones which occur less than 10 times in context) and rewrite them using a special unseen token. Features are then computed for all phones in the train set (including the special unseen token) using the procedure described in 2. At test time, all phones in unseen contexts are mapped to this special token and are represented by the corresponding feature vector.

4.3. Evaluation

We have evaluated the systems using both objective measure and subjective listening tests. In table 1, we present the results of the objective evaluation using the measure RMSE (Synthesized wavefiles along with Mean Absolute error and correlation can be found at http://goo.gl/lqrkNU). It can be seen that the proposed systems on both the audiobooks have the mean squared errors close to the complete systems FC_{CART} and FC_{NN} using hand tuned feature representations. Thus it can be seen that the proposed continuous representation indeed captures the required information to characterize the behavior of the phone and can be used to predict durations. However, although a quantitative understanding of the effects of our proposed duration modeling scheme is useful, it is more critical to perform subjective (listening) tests to understand the perceptual effects of the proposed duration modeling, since that is the most important metric we need to consider in building a TTS system. For this study, thirty test sentences were randomly selected from the audiobook data. 20 human listeners were asked to assign mean opinion scores (MOS) and relative preference scores to speech synthesized using these systems. We have followed the procedure for Blizzard listening tests [34] and also included a 'corrupt' set. That is, each experimental set consisted of the speech generated (from the same source text) by each of the aforementioned systems in addition to 'corrupted' speech generated by an additional system, where durations were set to unrealistically extreme values. (This system was included to catch instances of cheating and to ensure that the assigned ratings were realistic, and obviously excluded from the results reported).

When comparing the systems MD till VSM_{PPI} the improvement can be clearly seen both in terms of RMSE and MOS. This signifies the importance of using a suitable duration prediction module over mean durations. Comparing VSM_{PPI} with the systems FC_{CART} and FC_{NN}, it can be seen that the continuous representation based systems perform on par with the systems using hand tuned features.

5. Conclusions and Future work

We have presented a statistically trained continuous feature representation based framework for duration modeling. The system with mean durations was observed to be less effective both in terms of RMSE and MOS measures compared to the systems with predicted durations as expected. We also presented a method to simplify the training process and improve the representations based on matrix factorization and showed both qualitatively and quantitatively that this system performs comparable to the state-of-the-art front-end duration prediction systems. It might be interesting to study the prediction error as a function of the length of the feature vector. In the future we plan to develop VSM-based prediction modules for other important prosodic features such as intonation, phrase-breaks, etc.

6. References

[1] C. Mayo, R. A. Clark, and S. King, "Multidimensional scaling of listener responses to synthetic speech," 2005.

[2] H. Kumar, J. Ashwini, B. Rajaramand, and A. Ramakrishnan, "Mile tts for tamil and kannada for blizzard challenge 2013," in *Blizzard Challenge Workshop 2013*, 2013.

[3] H. Zen, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Hidden semi-markov model based speech synthesis." in *INTERSPEECH*, 2004.

[4] Y. Hifny and M. Rashwan, "Duration modeling for arabic text to speech synthesis." in *INTERSPEECH*, 2002.

[5] E. Moulines and F. Charpentier, "Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones," *Speech communication*, vol. 9, no. 5, pp. 453–467, 1990.

[6] J. P. Van Santen and X. Niu, "Prediction and synthesis of prosodic effects on spectral balance of vowels," in *Speech Synthesis, 2002. Proceedings of 2002 IEEE Workshop on*. IEEE, 2002, pp. 147–150.

[7] J. P. Van Santen, "Contextual effects on vowel duration," *Speech communication*, vol. 11, no. 6, pp. 513–546, 1992.

[8] D. H. Klatt and W. E. Cooper, "Perception of segment duration in sentence contexts," in *Structure and process in speech perception*. Springer, 1975, pp. 69–89.

[9] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, *Classification and regression trees*. CRC press, 1984.

[10] P. Taylor, A. W. Black, and R. Caley, "The architecture of the festival speech synthesis system," 1998.

[11] K. Sreenivasa Rao and B. Yegnanarayana, "Modeling syllable duration in indian languages using neural networks," in *Acoustics, Speech, and Signal Processing, 2004. Proceedings.(ICASSP'04). IEEE International Conference on*, vol. 5. IEEE, 2004, pp. V–313.

[12] J. Yamagishi, H. Kawai, and T. Kobayashi, "Phone duration modeling using gradient tree boosting," *Speech Communication*, vol. 50, no. 5, pp. 405–415, 2008.

[13] J. H. Friedman, "Multivariate adaptive regression splines," *The annals of statistics*, pp. 1–67, 1991.

[14] M. Riedi, "Modeling segmental duration with multivariate adaptive regression splines." in *EUROSPEECH*, 1997.

[15] O. Goubanova and S. King, "Bayesian networks for phone duration prediction," *Speech communication*, vol. 50, no. 4, pp. 301–311, 2008.

[16] A. Lazaridis, I. Mporas, T. Ganchev, G. Kokkinakis, and N. Fakotakis, "Improving phone duration modelling using support vector regression fusion," *Speech Communication*, vol. 53, no. 1, pp. 85–97, 2011.

[17] H. Elovitz, R. Johnson, A. McHugh, and J. Shore, "Letter-to-sound rules for automatic translation of english text to phonetics," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 24, no. 6, pp. 446–459, 1976.

[18] S. Lewis, K. McGrath, and J. Reuppel, "Language identification and language specific letter-to-sound rules," *Colorado Research in linguistics*, vol. 17, no. 1, pp. 1–8, 2004.

[19] R. Kuhn and J.-c. Junqua, "Method for letter-to-sound in text-to-speech synthesis," Feb. 22 2000, uS Patent 6,029,132.

[20] V. Pagel, K. Lenzo, and A. Black, "Letter to sound rules for accented lexicon compression," *arXiv preprint cmp-lg/9808010*, 1998.

[21] A. W. Black, K. Lenzo, and V. Pagel, "Issues in building general letter to sound rules," 1998.

[22] Z. Wang, S. M. Wong, and Y. Yao, "An analysis of vector space models based on computational geometry," in *Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 1992, pp. 152–160.

[23] W. R. Caid, S. T. Dumais, and S. I. Gallant, "Learned vector-space models for document retrieval," *Information processing & management*, vol. 31, no. 3, pp. 419–429, 1995.

[24] O. S. Watts, "Unsupervised learning for text-to-speech synthesis," 2013.

[25] T. Mikolov, M. Karafiát, L. Burget, J. Cernockỳ, and S. Khudanpur, "Recurrent neural network based language model." in *INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, September 26-30, 2010*, 2010, pp. 1045–1048.

[26] T. Mikolov, S. Kombrink, L. Burget, J. H. Cernocky, and S. Khudanpur, "Extensions of recurrent neural network language model," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, 2011, pp. 5528–5531.

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

[28] O. Levy and Y. Goldberg, "Neural word embedding as implicit matrix factorization," in *Advances in Neural Information Processing Systems*, 2014, pp. 2177–2185.

[29] C. Zhai and J. Lafferty, "A study of smoothing methods for language models applied to ad hoc information retrieval," in *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2001, pp. 334–342.

[30] K. Prahallad, "Automatic building of synthetic voices from audio books," Ph.D. dissertation, CMU, Pittsburgh, Jul., 2010.

[31] A. W. Black, "Clustergen: a statistical parametric synthesizer using trajectory modeling." in *INTERSPEECH*, 2006.

[32] A. Black, P. Taylor, R. Caley, R. Clark, K. Richmond, S. King, V. Strom, and H. Zen, "The festival speech synthesis system, version 1.4. 2," *Unpublished document available via http://www. cstr. ed. ac. uk/projects/festival. html*, 2001.

[33] G. K. Anumanchipalli, K. Prahallad, and A. W. Black, "Festvox: Tools for creation and analyses of large speech corpora," in *Workshop on Very Large Scale Phonetics Research, UPenn, Philadelphia*, 2011.

[34] A. Black and K. Tokuda, "The blizzard challenge 2005: Evaluating corpus-based speech synthesis on common databases," in *Proceedings of Interspeech*, 2005.