LETTER-TO-SOUND CONVERSION USING COUPLED HIDDEN MARKOV MODELS FOR LEXICON COMPRESSION

Hao Che[1]*, Jianhua Tao*[2]*, Shifeng Pan*[3]

[1,2,3]National Laboratory of Pattern Recognition,
Institute of Automation, Chinese Academy of Science, Beijing, China
{hche, jhtao,sfpan}@nlpr.ia.ac.cn

ABSTRACT

Letter-to-Sound(LTS) conversion, which is used to compress the lexicon for embedded application purpose, has become an important part in Text-to-Speech (TTS) system. In this paper, coupled Hidden Markov Models (CHMM) for LTS conversion is proposed. In the phase of preprocessing, many-to-many alignment is adopted for lexicon alignment instead of one-to-one alignment which is commonly used in previous approaches. Two Hidden Markov Models (HMM) which are respectively designed to predict the best phonemic string and corresponding graphemic substring segmentation are coupled in the phase of phonemes generation. The best phonemic string as the global optimal solution is given by maximizing the joint likelihood. Both combined and separated phone/stress prediction are concerned in stress assignment. The experimental result shows the performance of our approach is better than other previous approaches.

Index Terms—Letter-to-sound conversion, coupled Hidden Markov Models, many-to-many alignment, lexicon compression, stress assignment

1. INTRODUCTION

The performance of TTS synthesis system partially depends on the embedded lexicon. Usually, the size of lexicon is larger, the performance of TTS is better. However the size of lexicon is limited in some memory limited device with TTS system, such as GPS and mobile phone. Therefore letter-to-sound conversion is proposed to compress the size of lexicon, which could generate phonemic string according to the input word.

Many previous approaches tried to implement LTS conversion using orthographic rules which were written by hand in early years. Since orthographic conversion rules could only deal with simple input words, many data-driven approaches have been proposed [1]. The decision tree was proposed by Pagel and Black in 1998 [2]. They reported a word accuracy of 78% for OALD, 62% for CMUDict and 94% for BRULEX. The expectation maximization (EM) algorithm was applied to lexicon alignment in the paper. The PbA was employed for the Festival TTS synthesizer by Pamper in 2001 [3]. 86.7% word correct accuracy was obtained for OALD dictionary. Taylor proposed an approach for LTS conversion using HMM in 2005, where phones were hidden states and letters were observations [4]. The result of experiment showed a word accuracy of 61.08% was achieved. Not like above approaches mentioned which adopted the one-to-one alignment, the HMM which applied many-to-many alignment was proposed by Jiampojamarn in 2007 [5]. The experiments on CMUDict and other language lexicons showed that the approach could achieve better performance. An online discriminative training approach using many-to-many alignment was also presented by Jiampojamarn in 2009 [6]. This approach obtained a word accuracy of 72.5% for CMUDict, 89.6% for OALD.

However, as the phoneme was predicted for each letter independently without using other predictions from the same word, only phonemic context and relationship between phoneme and letter were used in previous approaches. Lack of letter context analysis was the shortcoming of these methods. Many recent studies show the accuracy of LTS conversion will benefit from involving letter context to phoneme generation model [7]. Since the HMM is suitable for capturing the interaction between letter and phoneme, CHMM which is consist of two HMMs for LTS conversion is proposed in this paper. One of HMM is designed to predict the best graphemic substring segmentation. The phoneme is considered as the states and the graphemic substring is considered as the observations in this HMM. In another HMM, which is used to generate the best phonemic string, the phoneme is observations and the graphemic substring is the states. Before generating phonemes all the reasonable graphemic substring segmentations are given. Then the best combination of phonemic string and graphemic substring segmentation is given by maximizing the joint likelihood of two HMMs. Both combined and separated phone/stress prediction are concerned in stress assignment. Our approach could obtain lexicon compression ratio of 74.6% for CMUDict and 94.2% for OALD in experiment, which is better than other approaches.

The rest of this paper is organized as follows. Section 2 reveals the many-to-many alignments approach used in our

approach. Our application of CHMM for LTS conversion is presented in section 3. In section 4 we evaluate our approach and compare the experimental results with other previous approaches. Section 5 discusses the remains problem and how to improve the performance in future.

2. GRAPHEME-TO-PHONEME ALIGNMENT

As the length of graphmeic string is generally not equal to the length of phonemic string for lexicon word, the alignment must be identified before a prediction model can be trained. Previous work generally assumed one-to-one alignment for simplicity. Since the length of graphemic string was not always equal to the length of corresponding phonemic string, epsilon (ε) was introduced to represent the "null" letter or phoneme in [9]. For example, a one-to-one alignment of word "exert" is showed in Tab.1 and "-" stands for epsilon:

Table 1. One-to-one alignment of word "exert"

Grapheme	e	x	-	e	r	t
Phoneme	ih	g	z	er	-	t

However, there are some problems with one-to-one alignment. It is difficult to decide which letter or phoneme should be aligned with epsilon for generation model. Another problem is that it is difficult for annotator to find the errors introduced by automatic alignment. The experimental results in [6] showed that the performance of the approaches using many-to-many alignment was better than the same approaches using one-to-one alignment.

Hence the many-to-many alignment approach is adopted to solve these problems in this paper. This approach assumed graphemic string with a length of not more than *maxX* could correspond to phonemic string with a length of not more than *maxY*. The EM algorithm was applied to estimate the probability of mapping graphemic string x^T to phonemic string y^V [5]. Variables T and V were the length of x and y respectively. For example, the word "watched", with phonemes [w aa ch t], is aligned as:

Table 2. Many-to-many alignment of word "watched"

Grapheme	wa	t	ch	ed
Phoneme	w	aa	ch	t

3. CHMM FOR LTS CONVERSION

3.1. Preprocessing

3.1.1. Symbol Definition

The CHMM could be decoupled into two HMMs. The HMM for predicting the best graphemic substring segmentation is denoted by λ^g. Another HMM for predict the best phonemic string is denoted by λ^p. The input graphemic string and output phonemic string are denoted by $G = \{g_1, g_2, g_3, \cdots, g_M\}$ and $P = \{p_1, p_2, p_3, \cdots, p_N\}$ respectively. G and P are observations and states sequence in λ^p respectively. The roles of G and P are exchanged in λ^g.

3.1.2. Graphemic substring segmentation

As one letter could be aligned with only one phoneme or epsilon, the concept of graphemic substring segmentation weren't concerned in previous LTS conversion approaches based on one-to-one alignment. Since one phoneme could be aligned with more than one letter in many-to-many alignment approaches, we have to consider all the possible graphemic substring segmentation in our approach.

Although the use of permutation and combination method is easy to list all the graphemic substring segmentations, we hope to deal with the segmentations that seem more reasonable, rather than all of them. For example, there are generally four boundary segmentations for word "use": {"u s e", "u se", "us e", "use"}. We could learn from phonetic that the string "use" shouldn't be considered because no phoneme could be aligned with "use". It is not only able to reduce the complexity of phoneme generation model, but also to avoid some unreasonable segmentaions confusing the final output. Therefore, some of them whose probability is too low should be rejected.

Both bigram model and trigram model could be used to calculate the corresponding probability $P(G)$. Concerning the complexity of the English phonetics, the trigram model is applied here. Then the probability of segmentation is:

$$P(G_j) = \prod_{i=1}^{n} P(g_i \mid g_{i-1}, g_{i-2}) \qquad (2)$$

The threshold θ is set to reject the unreasonable segmentations whose $P(G)$ is lower than θ. The θ is set to 0.25 according to experimental result.

3.2. Phoneme Generation Model

The performance of many previous studies using λ^p showed that λ^p was suitable for predicting phonemic string according to given graphemic string. Before that λ^g could be used to generate the best graphemic substring segmentation. However, in this way only suboptimal solution could be achieved for that these two processes are optimized separately. To get a global optimal solution the CHMM, which is described as $\lambda^g + \lambda^p$, could be applied.

We first analyze the implementations of λ^p and λ^g separately. λ^p is used to generate the phonemic string P corresponded to given G. This process can be expressed as:

$$P(G,P) = \prod_{i=1}^{n} P(g_i \mid p_i) P(p_i \mid p_{i-1}, p_{i-2}) \qquad (3)$$

$$P = \arg\max_{P} P(P \mid G)$$
$$\Downarrow$$
$$P = \arg\max_{P} \frac{P(G \mid P) \bullet P(P)}{P(G)} \quad (4)$$
$$\Downarrow$$
$$P = \arg\max_{P} P(P, G)$$

The Viterbi algorithm is applied to decode the optimal state sequence P^*. The process of λ^g is similar to λ^p except the likelihood function of *P(P,G)*:

$$P(G,P) = \prod_{i=1}^{n} P(p_i \mid g_i) P(g_i \mid g_{i-1}, g_{i-2}) \quad (5)$$

Then, the likelihood functions of λ^g and λ^p are combined in Equ.7:

$$\begin{aligned}(P,G) &= \arg\max_{P,G} \{\alpha\lambda^p + \beta\lambda^g\} \\ &= \arg\max_{P,G} \{\alpha \prod_{i=1}^{n} P(g_i \mid p_i) P(p_i \mid p_{i-1}, p_{i-2}) \\ &\quad + \beta \prod_{i=1}^{n} P(p_i \mid g_i) P(g_i \mid g_{i-1}, g_{i-2})\}\end{aligned} \quad (6)$$

The best combination of phonemic string and graphemic substring segmentation that is denoted by *(P,G)* could be determined by maximizing the joint likelihood. We can learn from the language model that the grapheme chunks segmentation isn't benefit from $P(p_i \mid g_i)$. On the contrary, some phoneme prediction errors are brought by $P(p_i \mid g_i)$. Therefore $P(p_i \mid g_i)$ is eliminated from Equ.7. Meanwhile, in order to reduce the computation cost, only parameter β is unknown and α is set to 1. So the Equ.7 is rewritten as:

$$\begin{aligned}(P,G) &= \arg\max_{P,G} \{\lambda^p + \beta\lambda^g\} \\ &= \arg\max_{P,G} \{\prod_{i=1}^{n} P(g_i \mid p_i) P(p_i \mid p_{i-1}, p_{i-2}) \\ &\quad + \beta \prod_{i=1}^{n} P(g_i \mid g_{i-1}, g_{i-2})\}\end{aligned} \quad (7)$$

The effect of λ^g and λ^p in LTS conversion could be modified by changing parameter β. According to phonetics we set the parameter β to 0.5 in actual calculation.

3.3. Stress Assignment

We can learn from the phonetics that the stress of the word could only be assigned to vowel. Therefore only the phone corresponding to vowel will be considered in stress prediction. The phoneme with stress is treated as new phonemes adding to phoneme generation models. For example, the possible phonemes set corresponding to the letter "a" in input word "able" are {"ah,ae,aa"} in phoneme generation model without stress prediction. The new set will double the amount of former set in stress prediction: {"ah,ah*,ae,ae*,aa,aa*"} (* is the stress phone).

There are two approaches to predict the stress: combined phone/stress prediction and separated phone/stress prediction. In combined prediction, both phones and stress are generated by a single model. On the contrary the stress is assigned by the independent HMM model in separated prediction. Both combined and separated prediction is tried in our experiment.

4. EXPERIMENT AND RESULTS

4.1. Lexicon

In order to compare with other approaches, CMUDict and OALD are selected as the input lexicon in our experiments. The CMUDict contains 105658 words along with their phonetic transcriptions and the OALD contains 72295 words. We split each lexicon into training set and test set based on the ratio of 9:1.

4.2 Lexicon alignment

How to find best values of *maxX* and *maxY* is an important work in lexicon alignment. Concerning the orthographic feature, the *maxY* was usually set to 2. So it is important to find suitable *maxX* decreasing the count of letter-phoneme pairs. Fig.1 shows the count of letter-phoneme pairs using different value of *maxX* in experiment. The experimental result shows that the count of letter-phoneme pairs is minimized while *maxX* and *maxY* are set to 4 and 2 respectively. For example, the word "caught", with phonemes [k aa t], could be aligned as[c→k, augh→aa, t→t].

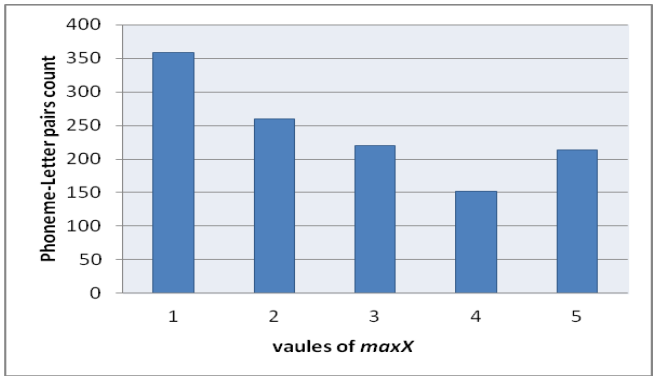


Fig. 1 Phoneme-letter pairs count according to different values of *maxX*.

4.3. Experimental Result

4.3.1. Comparing of combined and separated stress model

Since phone accuracy can't reflect the comprehensiveness of an approach, the word accuracy is selected to evaluate the

performance of models. Fig.2 shows the word accuracy of combined phone/stress prediction is higher than separated both for CMUDict and OALD, which is achieve 73.1% and 93.2% respectively.

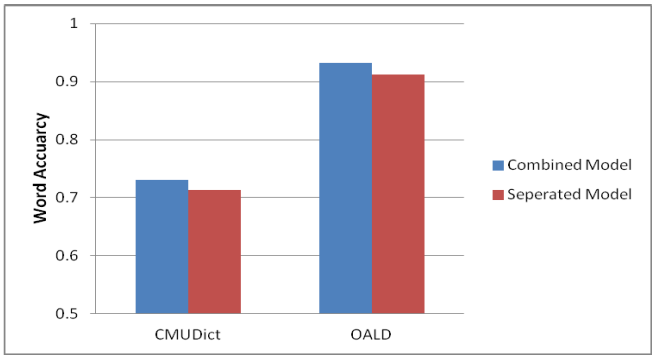
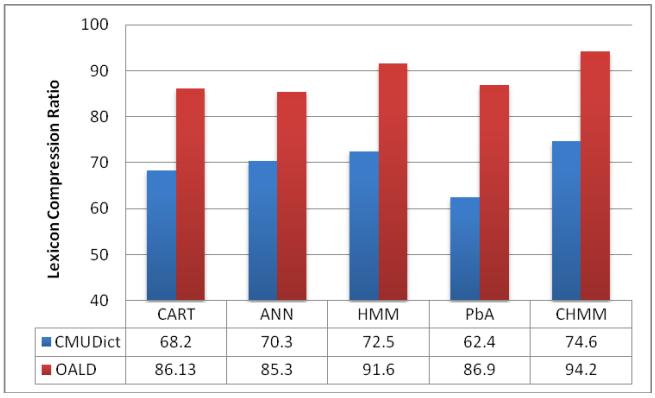


Fig. 2 Word accuracy of combined and separate phone/stress model for CMUDict and OALD.

4.3.2. Comparing with other approaches

For comparing with the performance of CHMM, some classic approaches are selected in experiment. Fig.2 shows the lexicon compression ratio of different approaches for CMUDict and OALD. The experimental result shows that the performance of CHMM is better than other approaches for either CMUDict or OALD.



	CART	ANN	HMM	PbA	CHMM
CMUDict	68.2	70.3	72.5	62.4	74.6
OALD	86.13	85.3	91.6	86.9	94.2

Fig. 3 Lexicon compression ratio of different approaches for CMUDict and OALD.

5. DISCUSSION

The experimental result shows that the performance of CHMM is better than previous approaches. However, many obvious errors could be found in result. For example, the phonetic transcripts of word “ostentation” and “ostentatious” are [ao s t eh n t ey sh ax n] and [aa s t ax n t ey sh ax s] separately. We notice that the vowel could be aligned with different phoneme in the same substring. For example, the vowel “o” could be aligned with “ao” and “aa” in substring “ostentatio”. As the same substring length is much larger than the different length of the substring, it is difficult for CHMM to generate correct phonemes. However, if the Part-of-Speech (POS) of word is involved, the model may have enough distinct features to distinguish the vowels in the same substring. So the POS is a latent factor to improve the accuracy of LTS conversion.

6. CONCLUSIONS

This paper proposes CHMM for LTS conversion. Instead of one-to-one alignment used in previous approaches, many-to-many alignment is applied to lexicon alignment. In order to integrate graphemic context and phonemic context in one model, two HMMs which are responsible for best phonemic string and best graphemic substring segmentation prediction separately are coupled in CHMM. The best phonemic string is generated in the sense of global optimization. Both combined prediction and separated phone/stress prediction are used in stress assignment. The experimental results show that the performance of our approach is better than other previous approaches.

7. ACKNOWLEDGEMENTS

The work was supported by NSFC-JSPS joint project (No.61011140075), the National Science Foundation of China (No.60873160 and No.90820303) and China-Singapore Institute of Digital Media (CSIDM).

8. REFERENCES

[1] R.I.Damper, Y.Marchand, M.J.Asamson and K.Gustafson, "A Comparison of Letter-to-Sound Conversion Techniques for English Text-to-Speech Synthesis", in Proceedings of the Institute of Acoustics, 20(6), pp. 245-254, 1999

[2] A.W.Black, K.Lenzo, and V.Pagel, “Issues in building general letter to sound rules”, in The Third ESCA Workshop in Speech Synthesis, pp. 77-80, 1998.

[3] Robert I.Damper, Craig Z.Stanbridge and Yannick Marchand, “A Pronunciation-by-Analogy Module for the Festival Text-to-Speech Synthesiser”, in Proc. 4th Int. Workshop Speech Synthesis, pp. 97-102, 2001.

[4] Paul Taylor, “Hidden Markov Models for Grapheme to Phoneme Conversion”, In Proceedings of INTERSPEECH, pp. 1973-1976, 2005.

[4] Sittichai Jiampojamarn, Grzegorz Kondrak and Tarek Sherif, “Applying Many-to-Many Alignments and Hidden Markov Models to Letter-to-Phoneme Conversion”, in HLT-NAACL 2007: Main Proceedings, pp. 372-379, 2007.

[5] Sittichai Jiampojamarn and Grzegorz Kondrak, “Online Discriminative Trainning for Grapheme-to-Phoneme Conversion”, in Proceeding of INTERSPEECH, pp. 1303-1306, 2009.

[6] Kheang Seng, Yurie Iribe and Tsuneo Nitta, “Letter-To-Phoneme Conversion based on Two-Stage Neural Network focusing on Letter and Phoneme Contexts”, in Proceeding of INTERSPEECH, pp. 1885-1888, 2011.

[7] Vincent Pagel, Kevin Lenzo and Alan W.Black, “Letter to Sound Rules for Accented Lexicon Compression”, in ICSLP, pp. 0561, 1998.