EuroSpeech, 3:1801-1804, Sept. 3-7, 2001, Aalborg, Denmark

A Two-Layer Lexical Tree Based Beam Search in Continuous Chinese Speech Recognition

Guoliang Zhang, Fang Zheng, and Wenhu Wu

Center of Speech Technology, State Key Laboratory of Intelligent Technology and Systems,
Department of Computer Science & Technology, Tsinghua University, Beijing, 100084, China
[liang, fzheng, wuwh]@sp.cs.tsinghua.edu.cn, http://sp.cs.tsinghua.edu.cn

Abstract

In this paper, an approach to continuous speech recognition based on a two-layer lexical tree is proposed. The search network is maintained by the two-layer lexical tree, in which the first layer reflects the word net and the phone net while the second layer the dynamic programming (DP). Because the acoustic information is tied in the second layer, the memory cost is so small that it has the ability to process some complicated applications, such as the use of cross-word context-dependent (CD) triphone models, the Chinese fuzzy syllable mapping and the pronunciation modeling. The search algorithm based on the two-layer lexical tree is also proposed, which is derived from the token-passing algorithm. Finally, an implementation of the two-layer lexical tree using the cross-word context-dependent triphone models is presented, and the experimental results show that the highly efficient decoding can be achieved without too much memory cost.

1. Introduction

A statistical method based speech recognition system is often composed of some essential parts: a Hidden Markov Model (HMM), a pronunciation dictionary, a search network and a language model, and the searching algorithm is also one of the key points. To construct a perfect recognition system, it is important to optimize the search algorithm in order to combine each component efficiently.

Recently, the lexical tree is used as the search network to reduce the search effort in many large vocabulary continuous speech recognizers [1]. The larger the vocabulary, the bigger improvement can we achieve. But the traditional lexical tree meets many difficulties in some new applications, for example:

First, the cross-word CD triphone models are widely used because of their good performance in modeling the co-articulation. However, the use of the cross-word CD triphone models with a static lexical tree results in a tremendous memory cost of the static lexical tree. Furthermore, the complete word conditioned lexical tree [2] becomes much bigger after integrating the language model. In some existing systems, the lexical tree has a great number of start nodes and end nodes with multiple looping-back re-entrant arcs [3][4].

Second, the Chinese fuzzy syllable set has been applied in a lot of Chinese recognition systems [5] in order to improve the robustness of processing many kinds of regional accents. In the fuzzy syllable set, many whole syllables need be mapped to other syllables, e.g. "*zhi→ji*", "guo→gui". For the lexical tree based on the CD triphone models, every fuzzy syllable in a word need expand at least one new branch. Therefore, the static lexical tree with many word influenced by the fuzzy syllable set trends to be more complicated and redundant.

Third, the research on the pronunciation variation modeling tends to be more and more important in speech recognition [6]. In Chinese speech recognition system, syllables, semi-syllables and phonemes are usually taken as the units for the pronunciation variation modeling. The similar difficulties will be met, that the memory cost of the static lexical tree grows too fast to accept when even only one syllable of every word has multiple pronunciation variations.

In the literature, a novel two-layer lexical tree is proposed to solve these problems. The traditional one-layer prefix lexical tree is designed as the first layer without being modified, and other information is integrated into the second layer. Because the information of the same recognition unit in different words is tied into one shared unit in the second layer, the memory cost is very small. It is widely understood that the organization of the search space for the dynamic programming is a key factor, so the two-layer lexical tree search will be efficient. Another advantage of the two-layer lexical tree is the good compatibility. The task of integrating the two-layer lexical tree into a new system is only to construct the second layer with a small effort easily.

The search algorithm based on the two-layer lexical tree is also presented in this paper, which is similar to the token-passing algorithm [7], a successful time-synchronous search algorithm. After combining with the language model, the search algorithm works very well in the large vocabulary Chinese speech recognition.

The rest of this paper is organized as follows. In Section 2, the framework of the two-layer lexical tree is given. In Section 3, the beam search algorithm based on the two-layer lexical tree is presented. In Section 4, we describe in detail the implementation of the two-layer lexical tree beam search to cope with the cross-word CD triphone models. Finally, some experiment results are given in Section 5 and conclusions are given in Section 6.

2. Two-layer lexical tree structure

For large vocabulary speech recognition, it is a very attractive idea to organize the pronunciation lexicon as a prefix tree [8]. In the lexical tree, all word beginnings are described as shared prefixes, which can reduce the search space greatly. Though a hypothesized word is only known when a leaf of the lexical prefix tree is reached, the look-ahead techniques [9] can shorten this gap effectively.

The search network can be sliced into three layers: the word net layer, the phone net layer and the DP net layer according to different knowledge sources [10]. The word net layer keeps the grammar constraints and specifications from the language model. The phone net layer keeps the phonetic lexicon graph

and so on. The DP net layer is based on the integration of all knowledge sources. The basic idea in this paper is adopting a two-layer lexical tree to maintain the three layers search network, where the first layer reflects the work net and the phone net while the second layer only influences the DP net.

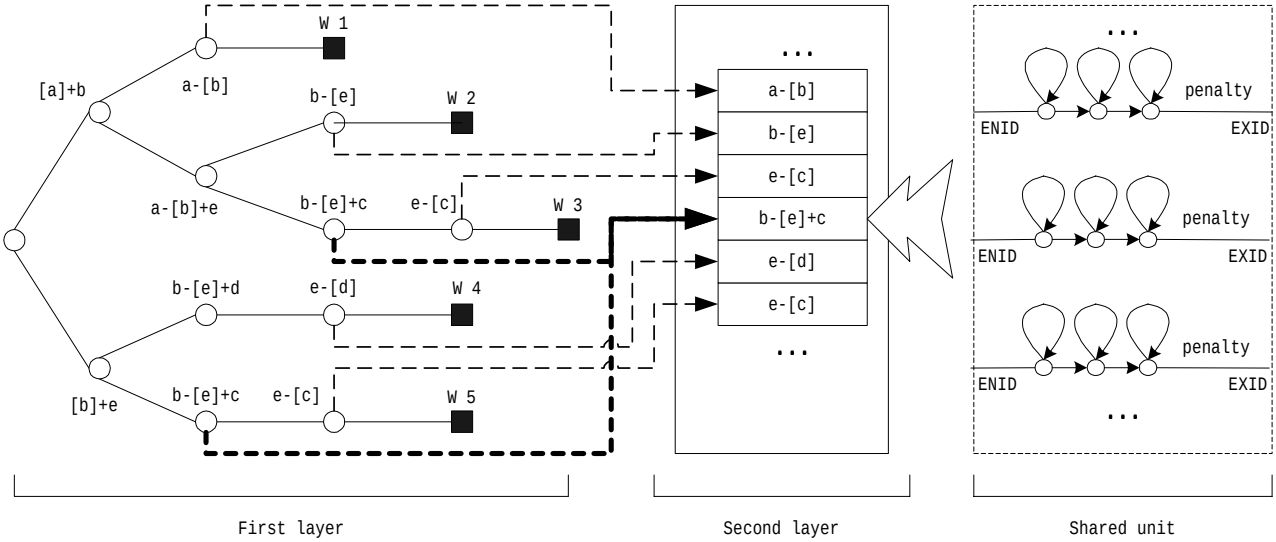

Figure 1. The two-layer lexical tree diagram
(A leaf node in the first layer contains the information of a word, i.e., the phone sequence from the root node to its parent node.)

The first layer of the lexical tree is independent of the acoustic model and does not carry any time information, deriving from the traditional prefix lexical tree based on the phoneme, the modification is replacing the acoustic model pointers with the pointers pointing to the shared units in the second layer. The second layer of the lexical tree reflects the DP net, so it need include a great deal of complicated information that has not been reflected in the first layer. The data structure of the second layer is an array of shared units, where every phoneme in the first layer corresponds to one and only one shared unit in which the model information and context information for the phoneme is shared. Because the information of the same phoneme in different words is tied into one shared unit, the memory cost of the second layer is very small. The framework of the two-layer lexical tree is illustrated in Figure 1.

The main task of the second layer is to construct the shared units which need to include some parallel arcs, indicating the propagating routes in the DP net of the two-layer lexical tree based search algorithm. In this paper, a *route* corresponds to one arc in the token. Four fields are assigned to an arc to memorize the arc information. The first field is an HMM pointer which records the acoustic model information. The second is the arc penalty that will be added to the score kept in the route traveling along the arc. The last two are one entry identity (ENID) and one exit identity (EXID), which are used in the search algorithm to indicate the left CD and right CD properties of the arc. If the acoustic model is a CD triphone model, the left and central phoneme identities are stored in the ENID while the central and right phoneme in the EXID. Otherwise, the ENID and EXID may be neglected.

The construction of the two-layer lexical tree is simple. Because the traditional lexical tree based on the phone can be converted into the first layer simply and the second layer is an array of shared units, the main task is to build a shared unit for each triphone model in the first layer of the lexical tree. We proceed as follows:

- Take the original triphone model as the first arc of the shared unit.
- Consider the variation of the central phoneme and insert a new arc that is made up of the left and right CD phoneme as well as the central phoneme variation.
- Consider the variation of the left CD phoneme and copy all the established arcs and replace the left CD phoneme with the variation.
- Consider the variation of the right CD phoneme and copy all the established arcs and replace the right CD phoneme with the variation.

Afterwards, the built shared unit in the second layer contains several arcs corresponding to all possible model variations of the triphone model. To clarify, a route is used in the search procedure to remember the current traveling status in an arc.

3. Search algorithm

In our approach, the framework of the two-layer lexical tree is still a pronunciation prefix tree, but the second-layer is separated, i.e. they are not integrated altogether in a single search-graph. Therefore, a modified search algorithm is needed.

Basically, we still use a time-synchronous beam search algorithm based on the token-passing algorithm. The token still accumulates the observation scores and refers to the trace back information. But instead of having only one route with a model pointer and an acoustic score, multiple routes are needed to incorporate the information in the second layer. The number of routes in a token is equal to the number of arcs in the token traveling shared unit.

The search scheme can be split into three steps according to the word net layer and the phone net layer. To formulate the dynamic programming approach, we define the following two quantities:

- $Q_v^p\left(t,s_y^x\right)$:= overall score of the best partial route stopping at node p in the first layer and at state s in the second layer whose ENID is x and EXID is y with the predecessor word v at time frame t.
- $B_v^p\left(t,s_y^x\right)$:= starting time of route $Q_v^p\left(t,s_y^x\right)$.

The first step is to process the intra-phone states propagating. Each route that propagates along an arc in the corresponding shared unit of the phone is independent of each other. This process can be formulated as following:

$$Q_v^p\left(t,s_y^x\right)=\max_{\sigma}\left\{q\left(x_t,s_y^x\mid\sigma_y^x\right)Q_v^p\left(t,\sigma_y^x\right)\right\} \quad (1)$$

$$B_v^p\left(t,s_y^x\right)=B_v^p\left(t,\max(\sigma_y^x)\right) \quad (2)$$

where $q\left(x_t,s_y^x\mid\sigma_y^x\right)$ is the product of the HMM transition and emission probabilities.

The second step is to process the inter-phone propagating. At the phone boundaries, the token continues to propagate along each arc in the following shared unit. Each arc chooses an active exit whose EXID equal to the ENID of the arc and uses the score of the corresponding active exit to propagate the token. We have to pass on the score and the time index before processing the hypotheses for time frame t:

$$Q_v^p\left(t-1,s_*^x=0\right)=\max_{\substack{z,f\\ f\in parent(p)}}\left\{Q_v^f\left(t-1,\sigma_x^z=end\right)\right\} \quad (3)$$

$$B_v^p\left(t-1,s_*^x=0\right)=B_v^{\max(f)}\left(t-1,\sigma_x^{\max(z)}=end\right) \quad (4)$$

where s_*^x indicates state s in all routes whose ENID is x and EXID is not cared.

The third step is to process the inter-word propagating. This scheme is similar to the previous step, except that the language model score needs to be accumulated into the overall score of the route. The process is showed as follows.

$$Q_w^{initial}\left(t-1,s_*^x=0\right)=\max_{v,z}\left\{p(w\mid v)Q_v^{final}\left(t-1,\sigma_x^z=end\right)\right\} \quad (5)$$

$$B_w^{initial}\left(t-1,s_*^x=0\right)=t-1 \quad (6)$$

Because the word network and the phone network are only reflected in the first layer of the lexical tree, all routes in one token share the same language model score. As we all know, the prefix lexical tree has a shortcoming that the exact word identity will not be known until a leaf of the lexical prefix tree is reached. In our approach, the language model look-ahead techniques and the phoneme look-ahead techniques [9] are adopted to solve this problem effectively.

We limit the search algorithm to the use of the bigram language model currently. The trigram language model can be combined into the search graph as well. So the two-layer lexical tree can be integrated into not only the one-pass integrated search algorithm but also the two-pass word-graph search algorithm [11].

4. Implementation

In this paper, the two-layer lexical tree beam search has been used to process the cross-word CD triphone models only and the implementation will be introduced in this section. The use of the lexical tree structure for the fuzzy syllable mapping and pronunciation modeling is not covered yet the integrating is limited to the second layer as well.

4.1. Speech recognition unit (SRU)

The Initial/Final (IF) structure is a particular characteristic of Chinese syllables. Mostly each Chinese syllable is consisting of an Initial and a Final, while some of them have only the Final part. The basic SRU set of our system is an extended Initial/Final (XIF) set with 27 Initials and 38 Finals, where 6 zero-Initials are added. The XIF set is showed in Table 1. The CD XIF model is adopted as the acoustic model in the system.

Table 1: The Extended Initial/Final Set

Type	Unit list
Initial (27)	*b, p, m, f, d, t, n, l, g, k, h, j, q, x, zh, ch, sh, z, c, s, r, _a, _o, _e, _I, _u, _v*
Final (38)	*a, ai, an, ang, ao, e, ei, en, eng, er, o, ong, ou, i, i1, i2, ia, ian, iang, iao, ie, in, ing, iong, iou, u, ua, uai, uan, uang, uei, uen, ueng, uo, v, van, ve, vn*

4.2. Two-layer lexical tree

The first layer of the lexical tree maintains the framework, in which each node denotes a CD XIF unit with the between-node links to describe the node relationship and a corresponding shared unit in the second layer to specify the acoustic model information. An Initial node and a Final node are added to the first-layer to form the looping back re-entrant arc.

The second-layer is an array of shared units. In our system, one word is consisting of a right-side CD XIF, some two-side CD XIF and a left-side CD XIF. To describe the information of the cross-word context-dependent models in the second-layer, the shared unit corresponding to the right CD XIF contains multiple arcs which cover all possible CD XIFs whose mid-part and right-part are same as the right CD XIF, likewise, the shared unit corresponding to the left CD XIF contains multiple arcs which cover all possible CD XIFs whose mid-part and left-part are same as the left CD XIF. If the pronunciation variation need not be considered, the shared unit corresponding to the CD XIF in the mid-part of a word only contains one arc.

4.3. Route pruning

Because the number of arcs in the shared unit may be large, the route pruning can be used to reduce the size of search effort without significantly increasing the word error rate. During the acoustic recognition process, only the most likely route in a shared unit hypotheses has to be retained at every time frame t. Therefore, we have to determine the score of the best route as follows.

$$R_v^p(t)_{\max}=\max_{s,x,y}Q_v^p\left(t,s_y^x\right) \quad (7)$$

The routes with scores relatively close to that of the best route will be kept active while others pruned. In formula, we have:

$$Q_v^p\left(t,s_y^x\right) < f_{ROUTE}\cdot R_v^p\left(t\right)_{\max} \quad (8)$$

5. Experiments

All experiments are based on a Mandarin dictation system, named *Easytalk*. Both the training and testing sets are taken from the "863" assessment [5], which contains 80 speakers' data and 520 utterances are available for each speaker. All the recorded materials are obtained in a low noise environment through a close-talk noise-canceling microphone at 16 kHz sampling frequency. 42-dimensional features are used for recognition. Each feature vector is made up of 13-dimensional MFCC and 1-dimensional log energy, their auto-regressive coefficients, and the 1st order derivatives of the auto-regressive coefficients. The CD XIFs are taken as the SRUs, with each modeled by a three-state HMM using HTK [12].

The training corpus includes about 36,400 utterances from 70 males. *Testing Set I* includes 1,000 utterances from 2 males while *Testing Set II* 240 utterances from 6 males. A two-layer lexical tree constructed from the 50K-word vocabulary with an average 4.8 XIFs per word. The baseline search strategy is the intra-word CD XIF model search strategy in which each shared unit of the two-layer lexical tree includes only one route with the phoneme variation not considered.

Table 2: Comparison on performance

	Intra-word search strategy	Cross-word search strategy	Word error reduction
Testing Set I	78.1%	92.3%	64.5%
Testing Set II	74.5%	88.5%	54.9%

We compare the two-layer lexical tree based cross-word CD XIF model search strategy with the baseline strategy. The experiment results given in Table 2 show that the average word error rates can be reduced by approximately 60%.

Table 3: Comparison on memory cost

	Intra-word search strategy	Cross-word search strategy
First layer	1.86MB	1.86MB
Second layer	0.31MB	0.46MB
DP	4MB	12MB

From Table 3, we can see that the memory cost of the static two-layer lexical tree is very small, and the peak DP search space memory usage is also small for a 50K-word vocabulary continuous speech recognition system. The memory cost of the second layer can be neglected compared with the first layer memory usage.

The time consumed in the cross-word CD XIF model search strategy is four times of that used in the intra-word CD XIF model search strategy without the route pruning. However, the cross-word CD XIF model search strategy becomes very efficient with the route pruning.

6. Conclusions

In this paper, a two-layer lexical tree based beam search for the large vocabulary speech recognition is presented. The two-layer lexical tree maintains the whole search network. The first layer of the lexical tree reflects the word net and the phone net while the second layer the DP net. Because the information of the same recognition unit in different words is tied into one shared unit in the second layer, the memory cost is so small that it has the ability to process some complicated applications, such as the use of cross-word CD triphone models, the Chinese fuzzy syllable mapping and the pronunciation modeling with small effort to construct the second layer. The two-layer lexical tree has the good compatibility, which can be integrated into a new system with a small effort. Finally, an implementation of the two-layer lexical tree using the cross-word CD triphone model is presented, and the experimental results prove that the search strategy can achieve high efficiency and good performance without too much memory cost.

7. References

[1] Ney.H, Haeb-Umbach.R, "Improvements in beam search for 10000-word continuous speech recognition". *Proceedings of ICASSP1992,* San Francisco, Vol1, p9-12

[2] Ortmanns.S, Ney.H, "A Comparion of Time Conditioned and Word Conditioned Search Techniques for Large Vocabulary Speech Recognition". *Proceedings of ICSLP1996*, Philadelphia, p.2091-2094

[3] Gauvain.J.-L, Lamel.L, "Speaker-independent continuous speech dictation". *Speech Communication,* 15:21-37, October 1994.

[4] Kris.D, Jacques.D, "A Static Lexicon Network Representation For Cross-word Context Dependent Phones", *EuroSpeech'97*

[5] Zheng, F., Song, Z.-J, and Xu, M.-X, "EASYTALK: A large-vocabulary speaker-independent Chinese dictation machine", *EuroSpeech'99*, Vol.2, pp.819-822, Budapest, Hungary, 1999

[6] Zheng, F., Song, Z.-J., Fung, P., Byrne, W., "Mandarin Pronunciation Modeling Based on CASS Corpus," *Sino-French Symposium on Speech and Language Processing, pp. 47-53, Oct. 16, 2000, Beijing*

[7] Odell.J.J, "The Use of Context in Large Vocabulary Speech Recognition". *PhD thesis*, University of Cambridge, U.K., March 1995

[8] Alleva.F, Huang.X.-D, "Improvements on The Pronunciation Prefix Tree Search Organization", *Proceedings of ICASSP96*, p133-136

[9] Ortmanns.S, Ney.H, "Look-ahead Techniques for Fast Beam Search", *Computer Speech and Language,* 2000, 14, p15-32

[10] Zhou.Q, Wu.C, "An Approach to Continuous Speech Recognition Based on Layered Self-Adjusting Decoding Graph". *Proceedings of ICASSP97,* V3, p1779-1782

[11] Ortmanns.S, Ney.H, "A Word Graph Algorithm for Large Vocabulary Continuous Speech Recognition", *Computer Speech and Language,* 1997, 11, p43-72

[12] Yong, S., Kershaw, D., Odell, J., Ollason, D., Valtchev, V. and Woodland, P., The HTK Book (for HTK Version 2.2), Cambridge University, 1999