AN EFFICIENT BEAM PRUNING WITH A REWARD CONSIDERING THE POTENTIAL TO REACH VARIOUS WORDS ON A LEXICAL TREE

Tsuneo Kato, Kengo Fujita, Nobuyuki Nishizawa

KDDI R& D Laboratories Inc.
User Interface Laboratory
2-1-15 Ohara, Fujimino-shi, Saitama 356-8502, Japan
tkato@kddilabs.jp

ABSTRACT

This paper presents an efficient frame-synchronous beam pruning for automatic speech recognition. With conventional beam pruning, hypotheses that have a greater potential to reach various words on a lexical tree are likely to be pruned out, since this potential is not taken into account. To make the beam pruning less restrictive for hypotheses with a greater potential and vice versa, the proposed method adds a reward as a monotonically increasing function of the number of reachable words from the node where a hypothesis stays on a lexical tree, to the likelihood of the hypothesis. The reward is designed not to collapse the ASR probabilistic framework. The proposed method reduces the processing time from 30% to 70% for grammar-based tasks. For a language-model-based dictation task, it also causes an additional reduction from the processing time of the beam pruning with the language model look-ahead technique.

Index Terms—pruning, frame synchronous beam search, lexical tree

1. INTRODUCTION

Automatic speech recognition (ASR) engines always demand fast search algorithms to use larger language models and more accurate acoustic models for expansion of their domains and coverage. Fast search algorithms are also needed for the engines to be embedded in mobile devices. In standard HMM-based ASR engines, search efficiency is enhanced in two ways. Firstly, the search space is hierarchically structured in a word level network and an HMM-state-level network which represents the words, and the HMM-state-level network is organized as a lexical tree [1]. Secondly, the hypotheses used to search for the best path on the lexical tree are effectively reduced by frame-by-frame pruning without word accuracy deterioration. The idea of basic beam and histogram pruning [2] is to retain the most promising hypotheses for further searches and exclude the rest with reference to their likelihoods. These methods basically function well by setting proper thresholds. However, the required number of hypotheses to bring out the best word accuracy is still excessive as the vocabulary size increases.

Various methods have thus been proposed to shorten the processing time [1-8]. Two-pass search algorithms are effective ways to introduce a detailed N-gram language model or a detailed acoustic model in a short processing time [3,4]. The language model look-ahead technique [1,5] significantly reduces the required number of hypotheses by incorporating a language model as early as possible into the search on the lexical tree. However, this powerful technique is not available in grammar-based tasks which do not use linguistic probabilities. Improvements of the beam search are mainly concerning adaptive controls of a dynamic beam width [6,7]. As an improvement focusing on properties of the lexical tree, equal-depth pruning with an optimization technique has been proposed in [8]. However, the performance is supposed to be unstable when the number of hypotheses is severely reduced, because the basis, i.e. the top of the likelihood in each depth is chosen from a limited number of hypotheses of a certain depth.

We propose an improved beam pruning which takes the property of the lexical tree into consideration. The property is that a hypothesis staying at a state close to the root of a lexical tree has a greater potential to produce various word hypotheses than one close to a leaf, and that pruning of a hypothesis close to the root generally has a more adverse impact on the accuracy of the resultant recognized word sequence than that of a hypothesis close to a leaf. Therefore, the proposed method eases the pruning condition for the hypotheses close to the root. Unlike the language model look-ahead, the proposed method is applicable to the grammar-based tasks. The proposed method is also applicable in combination with the language model look-ahead and/or the two-pass search algorithms.

The remainder of this paper is organized as follows. An analysis of the distribution of reachable words on a lexical tree, and the proposed method are described in Section 2. Experiments on the processing time (real time factor: RTF) and accuracy (word error rate: WER) in three recognition tasks are reported in Section 3, while the proposed method is concluded in Section 4.

2. BEAM PRUNING WITH A REWARD CONSIDERING THE POTENTIAL TO REACH VARIOUS WORDS ON A LEXICAL TREE

2.1. Distribution of HMM states in terms of the number of reachable words on a lexical tree

The lexical tree is formed by merging the common partial HMM state sequences from the beginning between words. A sample of the lexical tree is shown in Fig. 1. The potential to reach various words from a state in the lexical tree is quantifiable by the number of reachable words. As easily seen from Fig.1, a lexical tree comprises a small number of states with a great potential, and a vast number of those with a limited potential. We first investigated the distribution of HMM states in terms of the number of reachable words for the lexical tree of the 10k-word railway station name
task which we use in Section 3. Fig. 2 shows the histogram. The vertical axis is on a logarithmic scale. The HMM states reaching a single word, those reaching two words, those reaching three and those reaching four occupy 71%, 19%, 3.2% and 1.8%, respectively. The number of HMM states decreases rapidly from those of a single reachable word to more. On the other hand, a few HMM states close to the root have hundreds or thousands reachable words. An HMM state next to the root has the maximal number of 1,738 reachable words. Naturally, the hypotheses on the lexical tree comprise a small number of those with a great number of reachable words, and a vast number of those with few reachable words. As mentioned above, a pruned hypothesis with more reachable words impinges more than a pruned one with fewer reachable words on the word accuracy of the resultant word sequence. Therefore, we ease the pruning condition in relative terms for the few hypotheses with a great number of reachable words, and tighten it for the vast hypotheses with few reachable words.

2.2. Beam pruning with a reward as a function of the number of reachable words on a lexical tree

As a hypothesis advances on a path from root to leaf, the number of reachable words decreases monotonically, and is narrowed down to one after the hypothesis passes the last branching state in the lexical tree. Leveraging this property, a reward as a monotonically increasing function of the number of reachable words is tentatively added to the likelihood of the hypothesis for pruning. In addition, the monotonically increasing function is set to be zero when the number of the reachable words is one. The reward eases the pruning for the hypotheses closer to the root, while tightening it for the hypotheses closer to the leaves. Furthermore, this does not collapse the ASR probabilistic framework because the reward is always zero at the leaf HMM states unless homonyms or other words with that word as their prefix exist in the lexicon. (Note that the probabilistic framework is preserved by not adding the reward to the likelihood of a word hypothesis even when homonyms or other words with that word as their prefix exist.)

In a case of grammar-based recognition without linguistic probabilities, the score $S(h)$ for pruning of a hypothesis $h$ is given by

$$S(h) = L_a(h) + R(W(h))$$

(1)

where $L_a(h)$, $W(h)$ and $R(W)$ denote the accumulated acoustic likelihood, the number of reachable words of the hypothesis $h$ in the lexical tree and the reward as a function of the number of reachable words, respectively. Strictly speaking, the reachable words depend on the grammatical context. However, $W(h)$ was pre-computed approximately as the number of reachable words from a state in the lexical tree without considering the grammatical context here for simplification.

In a case of recognition based on a probabilistic language model, the score $S(h)$ is given by

$$S(h) = L_a(h) + w_{lm}[L_a + L_{lm}(h)] + R(W(h))$$

(2)

where $L_a$, $L_{lm}(h)$ and $w_{lm}$ denote the accumulated linguistic likelihood from the word at the beginning to the previous word, the
likelihood of language model look-ahead for the hypothesis \( h \) and the language model weight, respectively.

Considering the "long-tail" distribution of the HMM states shown in Fig. 2, we presume two types of monotonically increasing functions which fulfill \( R(I) = 0 \), here. One is \( A) \) a logarithmic function as:

\[
R(W) = a_{\log} \cdot \log(W)
\]

The other is \( B) \) an asymptotic exponential function converging on a value \( a_{\exp} \) as:

\[
R(W) = a_{\exp} \cdot \left( 1 - \exp\left( \frac{(W - 1)}{b_{\exp}} \right) \right)
\]

The \( a_{\log} \), \( a_{\exp} \) and \( b_{\exp} \) in the equations are constant values. The functions are shown in Fig. 3.

The pruning employed in this paper is a standard beam and histogram pruning proposed in [2]. For the beam pruning, the maximum of \( S(h) \) among all hypotheses is selected as the basis \( S_{\text{max}} \) every frame. Then, all the hypotheses are determined retained or pruned out according to whether the score \( S(h) \) falls within a predefined threshold \( f_B \) from the basis \( S_{\text{max}} \) or not. Hypotheses which fulfill the following equation are retained.

\[
S(h) \geq S_{\text{max}} - f_B
\]

The histogram pruning is to limit the number of retained hypotheses under a predefined number \( N_{\text{max}} \). To dispense with computationally expensive sorting of hypotheses likelihoods, all the hypotheses are classified into ranges of a histogram once, and the hypotheses from the upper ranges are retained until the total number of retained hypotheses exceeds \( N_{\text{max}} \). The beam pruning and histogram pruning are used in combination.

### 3. EXPERIMENTS

#### 3.1. Evaluation tasks, test sets and experimental setup

The proposed method was evaluated by three recognition tasks: an isolated word recognition task, a grammar-based short sentence task without linguistic probabilities, and a dictation task based on a probabilistic language model. The isolated word recognition task is of 10k-word railway station names in Japanese. The short sentence task is of a formulaic train connection inquiry. The grammar accepts the pattern, “From <<departure station>> to <<arrival station>>” in Japanese. The dictation task is a general mail dictation task without linguistic probabilities, and a dictation task based on a 30k-word trigram language model.

Test sets of the tasks were collected using a recorder on cellphones in various noise environments. The noise environments were 30 places where people often use cellphones, including railway terminal stations, suburban railway stations, station squares, offices, roadsides and shopping malls. The test set of the isolated word recognition task was 957 utterances made by 50 male and female speakers, respectively. The test set of the train connection task was 500 utterances of the same speakers. The test set of the mail dictation task was 389 utterances of typical sentences from business mails. This test set was collected in a silent environment.

The experimental conditions were as follows. A total of 38 dimensional acoustic features composed of the standard acoustic features of ETSI ES201108 with CMS and their first and second derivatives excluding power were extracted from speech sampled at 8.0 kHz. The decoder used speaker-independent tied-state triphone models. In the isolated word recognition and the short sentence tasks, context-free grammars (CFGs) without linguistic probabilities on word entries were used with a one-pass frame-synchronous beam search. In the mail dictation task, a trigram language model was used with a one-pass frame-synchronous beam search.

The coefficients \( a_{\log} \), \( a_{\exp} \) and \( b_{\exp} \) of the proposed functions were optimized to minimize WER by experimenting with a development set of the size equivalent to that of the test set under a tight pruning condition. The tight pruning condition set the beam width \( f_B \) at 140, and the maximum number of retained hypotheses \( N_{\text{max}} \) at 500. The processing time was measured on a PC with Intel Pentium 4 3.0 GHz.

#### 3.2. Results of an isolated word recognition task

Fig. 4 shows the averaged real time factor (RTF) and word error rates (WER) for the isolated word recognition task of 10k-word railway station names. Four lines represent the basic pruning that is the beam and histogram pruning described in Section 2.2 applied to the likelihood without adding a reward, \( A) \) the beam and histogram pruning with the reward given by a logarithmic function,
B) the beam and histogram pruning with the reward given by an asymptotic exponential function, and the equal-depth pruning [6]. The parameter of each line is the strength of the pruning. Actually, the maximal number of retained hypotheses $N_{max}$ was shifted with the beam width $f_B$ fixed at 140. The looser the pruning, the lower the WER value, but the longer the RTF.

The proposed method B) reached WER of 17.2% at RTF 0.180, while the basic pruning reached the same WER at RTF 0.255, which was a 30% reduction. In contrast, the proposed method A) made no improvement. The optimized coefficients were $a_{exp} = 0.90$ for the method A), $a_{exp} = 30.0$ and $b_{exp} = 11.0$ for the method B).

Two functions with the optimized coefficients are shown in Fig. 3. The ineffectiveness of the logarithmic function is due to the small change of the reward in the region of the small number of reachable words.

The equal-depth pruning in a simple implementation was worse than the others. It might be improved if the depth levels are tuned.

### 3.3. Results of a grammar-based short sentence task

Fig. 5 shows the RTF and WER for the grammar-based task of the formulaic train connection inquiry. The WER was calculated based on the 1,000 departure and arrival station names in the 500 utterances. Four lines represent the same as shown in Fig. 4. While the WER of the basic pruning gradually approached the minimal value, that of B) reached the minimal value of the basic pruning 21.3% at RTF 0.25, which meant an approximately 70% reduction. The proposed method B) achieved a 1% lower WER than the basic pruning for the minimal value. The optimized coefficients were $a_{exp} = 20.0$ and $b_{exp} = 4.0$ for the proposed method B). The proposed method A) and the equal-depth pruning did not show improvements for the task either.

### 3.4. Results of a language-model-based dictation task

Fig. 6 shows RTF and WER for the 30k-word mail dictation task. In this dictation task, the basic pruning includes the language model look-ahead technique [6]. This technique actually reduces the processing time to less than 1/10 from that without the look-ahead technique. The other lines also include the look-ahead technique as their baseline.

While the basic pruning reached a WER of 20.0% with RTF 0.60, the proposed method B) reached the WER value with RTF 0.46, which was a 23% reduction. The optimized coefficients were $a_{exp} = 46.0$ and $b_{exp} = 0.4$ for B). The proposed method A) and the equal-depth pruning made no improvement.

Though the reward given by the asymptotic exponential function with the language model look-ahead technique improved the efficiency from the basic pruning with the language model look-ahead, the effect was weaker than that of the grammar-based recognition tasks. We consider this to be due to the similarity of the effects achieved by the language model look-ahead technique and the proposed reward given on the likelihood of hypotheses. The look-ahead value also decreases monotonically as a hypothesis advances on a path from the root in the lexical tree because the look-ahead value uses the maximum value of the linguistic likelihood among the reachable words.

Viewed from another perspective, the monotonically decreasing reward along with a path on a lexical tree can be interpreted as a heuristic expected gain of the likelihood on the path from the current HMM state to a leaf state in the lexical tree.

### 4. CONCLUSIONS

To make the frame-synchronous beam search more efficient and reduce the processing time, we propose the introduction of a tentative reward considering the potential to reach various words from the HMM state of a hypothesis into pruning. The reward given by an asymptotic exponential function greatly reduced the number of hypotheses required to retain the maximal word accuracy in grammar-based tasks. The reward revealed 30-70% reduction in processing time for the grammar-based tasks without losing accuracy. In the language-model-based dictation task, it revealed an additional 23% reduction in processing time from the pruning with the language model look-ahead technique.

### 5. REFERENCES