Improvement of a Structured Language Model: Arbori-context Tree

Shinsuke MORI, Masafumi NISHIMURA, Nobuyasu ITOH

IBM Research, Tokyo Research Laboratory, IBM Japan, Ltd.
mori@trl.ibm.co.jp

Abstract

In this paper we present an extention of a context tree for a structured language model (SLM), which we call an arbori-context tree. The state-of-the-art SLM predicts the next word from a fixed partial tree of the history tree, such as two exposed heads, etc. An arbori-context tree allows us to select an optimum partial tree of a history tree for the next word prediction depending on the effectiveness in the similar way that a context tree selects the length of the history (n of n-gram). The experiment we conducted showed that the test set perplexity of the SLM based on an arbori-context tree (79.98) was lower than that of the SLM with a fixed history (101.56).

1. Introduction

In recent years, some structured language models (SLM) are proposed for purposes of spoken language understanding. In these models, words are predicted from left to right like in an n-gram model, but a sentence is not a simple word sequence but a tree whose leaves are labeled with a word. Thus the history referred for the prediction of the next word is not a word sequence but partial parse trees covering the preceding words. In a model for English [1], each word is predicted from two right-most exposed heads of the history tree. In a model for Japanese [2], each word is predicted from the words depending on it and the words depending on them. In both models the shape, including the depth and the width, of the history tree referred for next word prediction is always the same. There must be, however, some cases in which a large history tree is more informative for next word prediction and some other cases in which a small history tree is more effective because it does not suffer from a data-sparseness problem.

A variable memory length Markov model [3], represented by a context tree, is a variant of an n-gram model. In this model, the length of each n-gram is increased selectively according to an estimate of the resulting improvement in predictive quality. For example, it may happen that in case that the previous word is "I," a variable memory length Markov model does not distinguish the word before the previous word like a bi-gram model ($n = 2$), but if the previous word is "of," the same variable memory length Markov model uses the word before the previous word to help predict the next one like a tri-gram model ($n = 3$). The word three word before can also be checked out if it is considered to have some information about the next word to be predicted. Thus a variable memory length Markov model of the same size as an n-gram model has higher predictive power and it is smaller while achieving the same predictive power as an n-gram model. A variable memory length Markov model is a flexible n-gram model on a "linear history."

In this paper we present a flexible model on a "tree-structured history" for SLMs. As we mentioned above, in SLMs, the history at each step of word prediction is a sequence of partial parse trees. This can be regarded as a tree by adding a virtual root having the partial parse trees under it. This is called a history tree. In our model, based on a data structure which we call an arbori-context tree, the partial tree of the history tree referred for the next word prediction is enlarged selectively in arbitrary direction according to an estimate of the resulting improvement in predictive power. The experiment we conducted showed that the test set perplexity of the SLM based on an arbori-context tree was much less than the SLM based on two exposed heads.

2. Structure language model based on an arbori-context tree

In this section, first we explain a dependency grammar version of the SLM [1] and second we propose an arbori-context tree: context tree for a history tree.

2.1. Structure language model

In an SLM, each word in a sentence is predicted not only from the preceding word sequence but also from the partial parse trees covering it. Thus, the probability of a sentence $\boldsymbol{w} = w_1 w_2 \cdots w_n$ and a complete parse tree T is calculated as follows:

$$P(\boldsymbol{w}, T) = \prod_{i=1}^{n} P(w_i|\boldsymbol{t}_{i-1}) P(\boldsymbol{t}_i|w_i, \boldsymbol{t}_{i-1}), \quad (1)$$

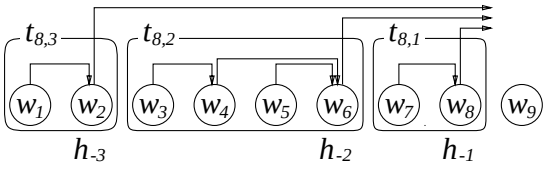

Figure 1: A partial parse tree.

where $\boldsymbol{t}_i$ is the i-th partial parse tree sequence. Figure 1 shows a situation before the 9th word prediction. In this figure, for example, first the 9th word is predicted from the 8th partial parse tree sequence $\boldsymbol{t}_8 = t_{8,3}t_{8,2}t_{8,1}$ and second the 9th partial parse tree sequence $\boldsymbol{t}_9$ is predicted from the 9th word and the 8th partial parse tree sequence $\boldsymbol{t}_8$ to get ready for the 10th word prediction. The problem here is to classify the condition part of the two conditional probability in the formula (1) in order to avoid a data-sparseness problem. In the paper presenting the SLM[1] the next word is predicted from two right-most exposed heads (for example w_8 and w_6 in Figure 1 as follows:

$$P(w_i|\boldsymbol{t}_{i-1}) \approx P(w_i|root(t_{i-1,2}), root(t_{i-1,1})),$$

where $root(t)$ is a function returning the root word of the tree t. A similar approximation is adopted to the probability function for the structure prediction. It is clear, however, that in some cases some child nodes of the tree $t_{i-1,2}$ or $t_{i-1,1}$ is informative for the next word prediction and in other cases even the distinction of an exposed head (root of the tree $t_{i-1,1}$ or $t_{i-1,2}$) causes a data-sparseness problem because of the limitation of the learning corpus size. Therefore a more flexible mechanism for history classification surely improves predictive power of the SLM.

2.2. Arbori-context tree

A variable memory length Markov model [3], an extension of n-gram model, is a flexible model for a linear history which selects the length of the history depending on the context. This model is represented by a tree whose nodes are labeled with a suffix of the context. This data structure is called a context tree. In this model, the length of each n-gram is increased selectively according to an estimate of the resulting improvement in predictive quality.

In SLMs explained above, the history is not a word sequence but a sequence of partial parse trees. This can be regarded as a single tree by adding a virtual root node having the partial trees under it. This is called a history tree. For example, Figure 2 shows the history tree for the 9th word prediction in Figure 1. The flexible mechanism for history tree classification we introduce in this paper is based on a data

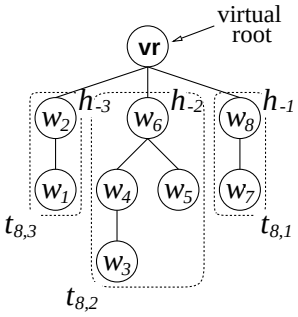

Figure 2: A history tree.

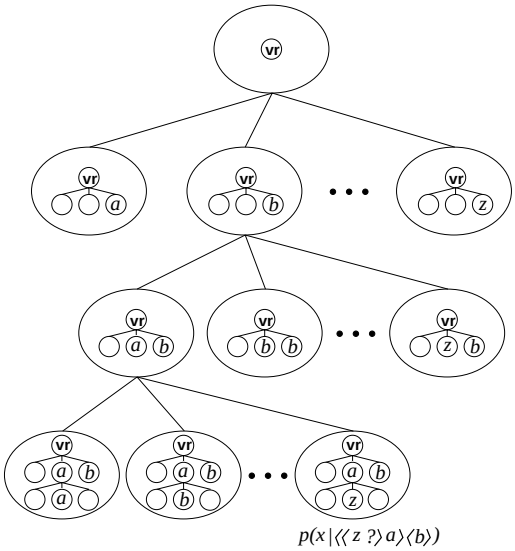

Figure 3: An arbori-context tree.

structure which we call an arbori-context tree. Each node of an arbori-context tree is labeled with a subtree of the history tree. The label of the root is the null tree and if a node has child nodes, their labels are the series of trees made by expanding a leaf of the tree labeling the parent node. For example, each child node of the root in Figure 3 is labeled with a tree produced by adding the right most child to the label of the root. Each node of an arbori-context tree has a probability distribution $P(x|t)$, where x is an alphabet and t is the label of the node. For example, let $\langle a_k \cdots a_2 a_1\rangle a_0$ represent a tree consisting of the root labeled with a_0 and k child nodes labeled with $a_k, \cdots, a_2, a_1$, the right most node at the bottom of the arbori-context tree in Figure 3 has a probability distribution of the alphabet x under the condition that the history matches the partial parse trees $\langle\langle z?\rangle a\rangle\langle b\rangle$. Putting it in another way, the

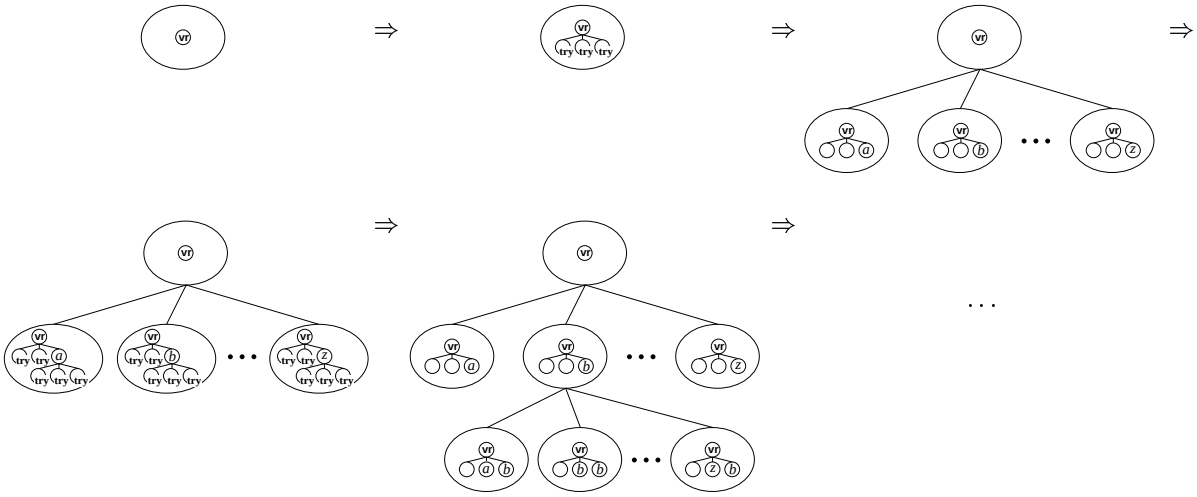

Figure 4: The process of an arbori-context tree creation.

next word is predicted from the history having b as the head of the right-most partial parse tree, a as the head of the second right-most partial parse tree, and z as the second right-most child of the second right-most partial parse tree. For example, in Figure 2 the subtree consisting of w_4, w_6, and w_8 is used for the prediction of the 9th word w_9 in Figure 1 if $a = w_6$, $b = w_8$, and $z = w_4$. Note that this is a specialization of the prediction from the two right-most exposed heads (w_6 and w_8). Thus in general, a model based on an arbori-context tree includes the model based on the two right-most exposed heads as its special case.

2.3. Creation of an arbori-context tree

The creation of an arbori-context tree is done as follows. In the beginning the tree has only a single node (root) labeled by the virtual node of the history tree. During the execution of the algorithm, nodes are added to the tree recursively as shown in Figure 4 according to the difference of the criterion function: the perplexity on the held-out corpus. The algorithm consists of two processes:

- select($leaf$) which calculates the difference of the criterion function of each possible expansion of the given $leaf$ and select the best way to expand the $leaf$. For example, if the argument is the central leaf of the 4th tree in Figure 4, the possible expansions are the specialization of the root of the second partial tree, the root of the third partial tree, the first child of the first partial tree, the second child of the first partial tree, and the third child of the first partial tree (where we suppose that the maximum length of the history tree sequence is 3). In Figure 4 the best way is the expansion of the root of the second partial tree.

- expand($leaf$, select($leaf$)) which try to expand the $leaf$ in the way given by select($leaf$) for all alphabet checking the criterion function and calls select($leaf$) recursively if some leaves are created or returns if there is no new leaf.

Table 1: Corpus.

	#sentences	#words	#chars
learning	7,677	218,628	336,726
test	853	24,268	37,552

3. Evaluation

In this section, we present the result of the experiments we conducted and evaluate our new flexible mechanism for history tree classification.

3.1. Conditions on the Experiments

The corpus used in our experiments consists of articles extracted from a financial newspaper in Japanese

Table 2: Test set perplexity of each model.

language model	word prediction	structure prediction	total (product)
SLM based on an arbori-contex tree	62.34	1.28	79.98
SLM based on two exposed heads	79.17	1.28	101.56
word tri-gram model	74.93	—	74.93

(Nihon Keizai Shimbun). Each sentence in the articles is segmented into words and annotated with a dependency structure by linguists at our site. The corpus was divided into ten parts; the parameters of the model were estimated from nine of them and the model was tested on the rest. Table 1 shows the corpus size.

To evaluate the predictive power of the SLM based on an arbori-context tree in comparison with the SLM based on two exposed heads, we constructed these models using the same learning corpus and calculated their perplexity on the same test corpus. In this process, the annotated tree in the test corpus is used as the structure of the sentences. Therefore the probability of each sentence in the test corpus is not the summation over all its possible derivations. Note that the component for structure prediction of the SLMs are common, thus the perplexity of this part is constant. We also calculated the test set perplexity of a word tri-gram model estimated form the same learning corpus. Unknown words are represented by part-of-speech symbols. Thus, the probability for their character generation is excluded.

3.2. Evaluation

Table 2 shows the test set perplexity of each model. The test set perplexity of the SLM based on an arbori-context tree is much less than that of the SLM based on two exposed heads. The reduction ratio is 21.25%. Since the components for structure prediction of the SLMs are common, the reduction ratio of the total test set perplexity is also 21.25%. This result attests experimentally that using an arbori-context tree we have succeeded in improvement of the SLM based on two exposed heads. The idea of arbori-context tree is so general that we can apply this flexible mechanism to the original SLM for English[1]. In comparison with the word tri-gram model, the SLM based on an arbori-context tree has much less test set perplexity if we exclude the contribution of the structure prediction component, and if we include it, the test set perplexity of the SLM is slightly higher than that of the word tri-gram model. This means that replacing the LM (word tri-gram model) of a speech recognizer by an SLM based on an arbori-context tree we can obtain a speech recognizer which outputs a word sequence as well as its structure (parse tree) without serious decrease in recognition accuracy. Therefore, an arbori-context tree is able to be a base of an effective language model from the viewpoint of spoken language understanding.

4. Conclusion

In this paper we have presented an arbori-context tree, a context tree on tree-shaped history (history tree). The state-of-the-art structured language models predict the next word from a fixed part of the history tree, such as two right-most exposed heads in [1] or the words depending on the next word and the words depending on them [2]. The data structure we have presented in this paper allows us to select a partial tree of a history tree in order to better predict the next word in a similar way that a context tree selects the length of the history string (n in n-gram model) depending on the effectiveness of the history.

The experiments we conducted showed that the test set perplexity of the structured language model based on an arbori-context tree (79.98) was lower than that of the structured language model with a fixed history (101.56). It follows that our new data-structure improves a structure language model in prediction power.

5. References

[1] Ciprian Chelba and Frederick Jelinek. Structured language modeling. Computer Speech and Language, 14:283–332, 2000.

[2] Shinsuke Mori, Masafumi Nishimura, Nobuyasu Itoh, Shiho Ogino, and Hideo Watanabe. A stochastic parser based on a structural word prediction model. In Proceedings of the 18th International Conference on Computational Linguistics, pages 558–564, 2000.

[3] Dana Ron, Yoram Singer, and Naftali Tishby. The power of amnesia: Learning probabilistic automata with variable memory length. Machine Learning, 25:117–149, 1996.