A THESAURUS-BASED STATISTICAL LANGUAGE MODEL FOR BROADCAST NEWS TRANSCRIPTION

Akio Ando, Akio Kobayashi and Toru Imai

NHK (Japan Broadcasting Corporation)
Science and Technical Research Laboratories,
1-10-11 Kinuta Setagaya Tokyo 157-8510 JAPAN
E-mail : {ando, akio, imai}@strl,nhk.or.jp

ABSTRACT

This paper describes a thesaurus-based class n-gram model for broadcast news transcription. The most important issue concerned with class n-gram models is how to develop a word classification. We construct a word classification mapping based on a thesaurus so as to maximize the average mutual information function on a training corpus.

To examine the effectiveness of the new method, we compare it with two our previous methods, in which the same thesaurus is used but word-class mappings are determined in different manners. The new method achieved substantially lower perplexity for 83 news transcription sentences broadcast on June 4, 1996.

1. INTRODUCTION

Class n-gram models have been studied for smoothing statistical language models by collecting statistics not on individual words, but rather on classes of words[1]. How to develop a word classification is the most important issue for such models. Brown et. al. proposed a method of developing a hierarchical word classification automatically from a text corpus[2]. It searches for the classification that minimizes the average conditional entropy on a training corpus. The problem with the method is that it cannot be directly applied to the large vocabulary case due to computational burden. To avoid the problem, they adopted a step-wise method in which pairs of words or classes are iteratively merged to construct a word classification based on the entropy measure.

On the other hand, the thesaurus is well known as a word classification based on human knowledge. We use it to improve the word classification. It usually has a tree structure which is a great advantage for hierarchical word classification. There are, however, two essential problems when applying the thesaurus to word classification: one is the problem of words that belong to more than one class, and the other is the problem of words that are not registered in the thesaurus. We solve these problems using the average mutual information function. Our algorithm hierarchically classifies each word based on the tree structure of the thesaurus. At the first stage, it searches nodes on the level immediately following the root node and classifies words into the classes corresponding to the nodes. Then it iteratively subdivides the classes by searching the tree to get the final result.

2. WORD CLASSIFICATION

2.1. Definition of word set and class

We choose high frequency words from a training corpus to build a vocabulary for broadcast news transcription. The size of the vocabulary in the experiment shown later is 20,000. We construct a word dictionary by adding to the vocabulary a word "UNK", which represents words not included in the vocabulary. Let V denote a set of words of the dictionary. Let V_1 denote a set of words included both in the thesaurus and the dictionary, and V_2 denote a set of words which are not included in the thesaurus but are included in the dictionary.

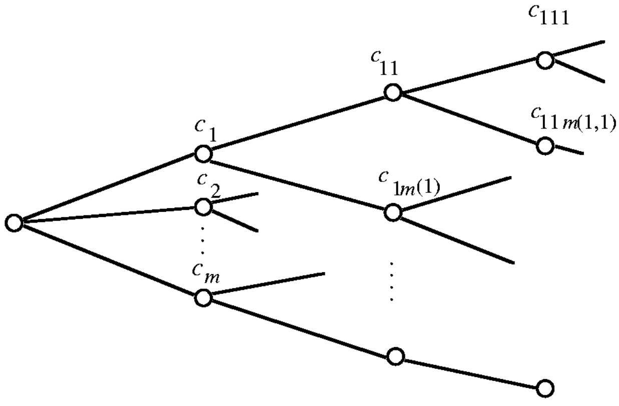

Figure 1: Tree structure of thesaurus

Therefore V is a union of V_1 and V_2.

Let C denote a set of classes and C_i denote a subset of C. C_i corresponds to the nodes directly following the root node in a thesaurus tree. Similarly, let $c_{i\cdot\cdot jk}$ denote a subset of $c_{i\cdot\cdot j}$(See Figure 1).

2.2 Objective function

A word classification mapping is constructed based on a bigram model. We use the average mutual information function as the objective function. Let F denote the classification function which maps V into C and let w_1^T represents the word string $w_1 w_2 ... w_T$. Then the probability of a string of words w_1^T with the bigram model can be expressed as a product of conditional probabilities:

$$P\left(w_1^T\right) = P(w_1)P(w_2|w_1)P(w_3|w_2)\cdots P(w_T|W_{T-1}) \quad . \qquad (1)$$

We assume that the best model is the model which maximizes (1). From the point of view of class language models, the value of $P(w_1)$ does not depend on the models. Then we maximize the average log probability:

$$L(F) = (T-1)^{-1}\log\left[P(w_2|w_1)P(w_3|w_2)\cdots P(w_T|w_{T-1})\right]. \quad (2)$$

We set (2) as a function of F because we want to get the optimal word classification mapping. We have

$$\begin{aligned}
&L(F)\\
&= (T-1)^{-1}\log\left[P(w_2|w_1)\cdot P(w_3|w_2)\cdots P(w_T|w_{T-1})\right]\\
&= (T-1)^{-1}\sum_{i=2}^{T}\log\left[P(w_i|w_{i-1})\right]\\
&= (T-1)^{-1}\sum_{i=2}^{T}\log\left[P(c_i|c_{i-1})\cdot P(w_i|c_i)\right]\\
&= (T-1)^{-1}\sum_{w_j w_k} S(w_j w_k)\log\left[P(c_k|c_j)\cdot P(w_k|c_k)\right]\\
&= (T-1)^{-1}\sum_{w_j w_k} S(w_j w_k)\log\left[\frac{P(c_k|c_j)}{P(c_k)}\cdot P(w_k|c_k)P(c_k)\right]\\
&= (T-1)^{-1}\sum_{c_j c_k} S(c_j c_k)\log\left[\frac{P(c_k|c_j)}{P(c_k)}\right] \qquad (3)\\
&\quad + (T-1)^{-1}\sum_{w_k}\left(\sum_{w_j} S(w_j w_k)\right)\log\left[P(w_k|c_k)P(c_k)\right] \quad ,
\end{aligned}$$

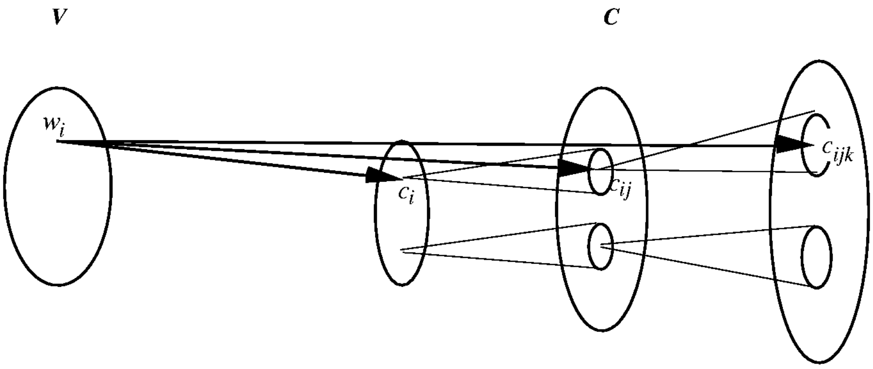

Figure 2: Hierarchical word classification

$$\boldsymbol{c_i = \{\ c_{i1}\ ,\cdots, c_{ij}\ ,\cdots, c_{jm(i)}\}}$$
$$\boldsymbol{c_{ij} = \{\ c_{ij1}\ \cdots, c_{ijk}\ \ \cdots, c_{ijm(i,j)}\ \}}$$

where $c_i \equiv F(w_i)$ and $S(w_j w_k)$ represents the number of times that the word pair $w_j w_k$ occurs in the string w_1^T. We must have, in the limit ($T \to \infty$),

$$\begin{aligned}
&L(F)\\
&= \sum_{w_j w_k} P(c_j c_k)\log\left[\frac{P(c_j c_k)}{P(c_j)P(c_k)}\right] + \sum_{w_k} P(w_k)\log\left[P(w_k)\right] \qquad (4)\\
&= I(c_j, c_k; F) - H(w) \quad ,
\end{aligned}$$

where $I(c_j, c_k; F)$ is the average mutual information function. Since the entropy H(w) is independent of any equivalence classification, maximizing $L(F)$ is equivalent to maximizing $I(c_j, c_k; F)$. Therefore we select $I(c_j, c_k; F)$ as the objective function.

2.3. Maximization of the objective function

Maximization of $I(c_j, c_k; F)$ is a kind of combinatorial optimization problem. Although various methods have been proposed in the area, we adopt a simple method mainly because of computational burden.

Our method is a hierarchical method based on the tree structure of the thesaurus. The classification mapping F is automatically formed by multiple-stage procedures. Each word in V is mapped to each of classes C_1, C_2,...,C_m in the first stage. In the following stages, each word which was mapped to the class $C_{i..j}$ is mapped to each of the subdivided classes $C_{i..j1}$, $C_{i..j2}$,..., $C_{i..jm(i..j)}$ (See Figure 2).

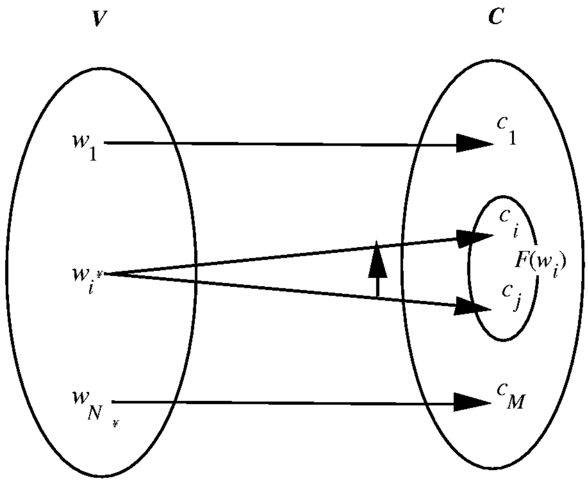

Figure 3: Perturbation

in V_1 is mapped to a uniquely determined class in the first step, and then each word in V_2 is mapped to an appropriate class in the second step.

Many words in V_1 usually belong to more than one class in the thesaurus. Therefore, we must select the most likely class for each word in V_1 to construct a mapping:

(Step a)
(a-1) Select a word *w* from V_1,
(a-2) Select a class *c* from among the classes to which *w* belongs (that is $F(w)=c$) so as to maximize the objective function on the training corpus (See Figure 3),
(a-3) Fix the value of $F(w)$ as the selected class in (a-2),
(a-4) If all words in V_1 have already been selected, then stop. Otherwise, select another word from V_1 and go to (a-2).

The second step extends the mapping with the same criterion:

(Step b)
(b-1) Initial condition:
$F(w)$ is decided in an appropriate manner. Calculate the initial value of the objective function. Set the value as *X*.
(b-2) Iteration:
(b-2-1) Select a word *w* from V_2,
(b-2-2) Select a class $c_{i \cdot\cdot jk}$ from $c_{i \cdot\cdot j}$, which was selected at the previous stage, so as to maximize the objective function on the training corpus (Also see Figure 3),
(b-2-3) Fix the value of $F(w)$ as the selected class in (b-2-2),
(b-2-4) If all words in V_2 have already been selected, then go to (b-2-5). Otherwise, select another word from V_2 and go to (b-2-2).
(b-2-5) Set the final value of the objective function as *X'*. If *X'*-*X* is less than the predetermined threshold, then stop. Else, substitute the value of *X'* for *X* and go to (b-2).

Through these two steps, we get the desired mapping which is expected to maximize the objective function.

In (Step b) of the algorithm, we automatically assign unknown words, which are not included in *V*, to the appropriate class.

3. TRAINING THE CLASS LANGUAGE MODEL

A class n-gram language model is trained using the word classification mapping. Each word of the training corpus is mapped to a class to get a class code sequence. Class n-gram models $\left\{P\left(c_k \middle| c_{k-1} \cdot\cdot c_{k-p}\right)\right\}$ and $\left\{P\left(w_k \middle| c_k\right)\right\}$ are calculated based on the sequence. In the experiment in the next section, we constructed a class trigram model.

4. EVALUATION

4.1 Test corpus and thesaurus

We used a corpus of NHK(Japan Broadcasting Corporation) news scripts broadcast in the period from April 1, 1991 to June 3, 1996 for constructing the mapping and training a class n-gram model. We selected the most frequent 20,000 words from the training set and treated the other words as unknown words. We used a Japanese thesaurus "Bunrui-goihyo" developed by the National Language Research Institute of Japan, which includes 87,743 words.

4.2 Evaluation experiment

To examine the effectiveness of the new method, we compared it with two methods in which the same thesaurus is used, but word-class correspondences are determined using a heuristic scheme and a dynamic programming strategy[3].

The thesaurus which we used has a tendency that important words are assigned to small code numbers. Thus we map each element (word) of V_1 to a class which has the smallest code number among the classes corresponding to the element. We call this a heuristic method. In the heuristic method, each element of V_2 is mapped to the newly defined class that corresponds only to the element. The other method determines word-class correspondence with a dynamic programming procedure. Figure 4 shows the strategy of the method. At the training stage, each $S(c_i c_j)$, the number of occurrences of class pair c_i and c_j, is calculated. For example, in the case of Figure 4, the numbers of occurrences for class pairs (c_{11},c_{12}), (c_{11},c_{22}), (c_{21},c_{31}), (c_{21}, c_{32}) ,..., ($c_{n\text{-}1,3}$, $c_{n,1}$) are added to $S(c_{11},c_{12})$, $S(c_{11},c_{22})$, $S(c_{21},c_{31})$, $S(c_{21}, c_{32})$,..., $S(c_{n\text{-}1,3}$, $c_{n,1})$ to get $\{p_{ij}^{(k)}\}$. Those probabilistic parameters are then used to select the optimal class sequence in the selection stage. The method maps each element of V_2 in the same manner as the heuristic method. Note that this word-class correspondence is not a mapping because there are some words which are mapped to more than one class.

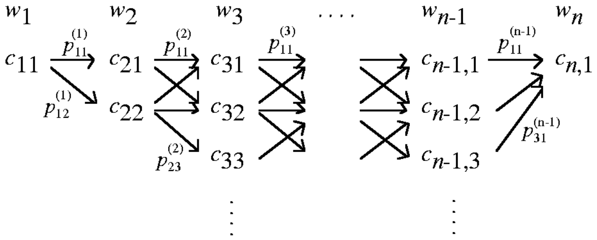

Figure 4: Selection of classes in our old method

$w_1, w_2, \cdots$: word sequences
$c_{31}, c_{32}, \cdots$F the classes corrensponding
to the word w_3
$p_{11}^{(1)}, \ldots$: joint probabilities among two classes

Table 1: Test set perplexity for three method

method	heuristic method	DP method	new method
trigram perplexity	213	219	149

We converted a word sequence from the training corpus to the class code sequence by each method for calculation of the class models. Test sentences used in the experiment were 83 news transcription sentences broadcast on June 4, 1996. We calculate perplexity values for the sentences with three models and compare these results.

4.3 Experimental result

We set the number of stages in the hierarchical construction of the classification mapping to 4. The number of classes was 493 for the new method. For the other methods, the number of classes was 780 because the classes corresponding to the elements of V_2 are added to the original classes.

Table 1 shows the trigram perplexity values for the test set by the three methods. The new method achieved substantially lower perplexity than the other two methods for the test set. The method with DP shows the highest perplexity. The word class correspondence created by the method does not satisfy the condition of mapping. Some words correspond to different classes in different contexts. That is thought to be the reason.

5. CONCLUDING REMARKS

This paper proposed a new method to construct a word classification mapping for class language models. The class language model with the new method achieved substantially lower perplexity for 53 news transcription sentences broadcast on June 4, 1996 in comparison with our previous methods.

For broadcast news transcription, word based statistical language models achieved better performance than class models especially for anchors' speech which consists of rather more stereotyped sentences than does conversational speech or interview speech. The advantage of using class models results from the smoothing method in which class models compensate for the sparsity problem of word-based language models. It is also applicable to models for conversational speech, for which few transcriptions exist. In recent broadcast news programs, conversations show a tendency to increase and should be covered by class language models. These applications of class language models are now under investigation.

6. REFERENCES

[1] F. Jelinek: "Self-organized language modeling for speech recognition", in *Readings in Speech Recognition*, A. Waibel and K.F.Lee, eds., pp.450-506, Morgan-Kaufmann 1990

[2] P.F.Brown, et. al.: "Class-based n-gram Models of Natural Language", Computational Linguistics, Vol.28, no.4 1992

[3] A. Ando, et. al.: "A Study of Statistical Language Model using Thesaurus-based Semantic Class", Proc. of Acoust. Soc. Japan 2-1-8, September 1997