Maximum Entropy Model Learning of the Translation Rules

Kengo Sato and **Masakazu Nakanishi**
Department of Computer Science
Keio University
3–14–1, Hiyoshi, Kohoku, Yokohama 223-8522, Japan
e-mail: {satoken,czl}@nak.ics.keio.ac.jp

Abstract

This paper proposes a learning method of translation rules from parallel corpora. This method applies the maximum entropy principle to a probabilistic model of translation rules. First, we define feature functions which express statistical properties of this model. Next, in order to optimize the model, the system iterates following steps: (1) selects a feature function which maximizes log-likelihood, and (2) adds this function to the model incrementally. As computational cost associated with this model is too expensive, we propose several methods to suppress the overhead in order to realize the system. The result shows that it attained 69.54% recall rate.

1 Introduction

A statistical natural language modeling can be viewed as estimating a combinational distribution $X \times Y \to [0,1]$ using training data $\langle x_1, y_1 \rangle, \ldots, \langle x_T, y_T \rangle \in X \times Y$ observed in corpora. For this topic, Baum (1972) proposed EM algorithm, which was basis of Forward-Backward algorithm for the hidden Markov model (HMM) and Inside-Outside algorithm (Lafferty, 1993) for the probabilistic context free grammar (PCFG). However, these methods have problems such as increasing optimization costs which is due to a lot of parameters. Therefore, estimating a natural language model based on the maximum entropy (ME) method (Pietra et al., 1995; Berger et al., 1996) has been highlighted recently.

On the other hand, dictionaries for multilingual natural language processing such as the machine translation has been made by human hand usually. However, since this work requires a great deal of labor and it is difficult to keep description of dictionaries consistent, the researches of automatical dictionaries making for machine translation (translation rules) from corpora become active recently (Kay and Röschesen, 1993; Kaji and Aizono, 1996).

In this paper, we notice that estimating a language model based on ME method is suitable for learning the translation rules, and propose several methods to resolve problems in adapting ME method to learning the translation rules.

2 Problem Setting

If there exist $\langle x_1, y_1 \rangle, \ldots, \langle x_T, y_T \rangle \in X \times Y$ such that each x_i is translated into y_i in the parallel corpora X, Y, then its empirical probability distribution $\tilde{p}$ obtained from observed training data is defined by:

$$\tilde{p}(x,y) = \frac{c(x,y)}{\sum_{x,y} c(x,y)} \tag{1}$$

where $c(x,y)$ is the number of times that x is translated into y in the training data.

However, since it is difficult to observe translating between words actually, $c(x,y)$ is approximated with equation (2) for sentence aligned parallel corpora.

$$c(x,y) = \sum_i \frac{c'_i(x,y)}{|X_i||Y_i|} \tag{2}$$

where X_i is i-th sentence in X. We denote that sentence X_i is translated into sentence Y_i in aligned parallel corpora. And $c'_i(x,y)$

is the number of times that x and y appear in the i-th sentence.

Our task is to learn the translation rules by estimating probability distribution $p(y|x)$ that $x \in X$ is translated into $y \in Y$ from $\tilde{p}(x, y)$ given above.

3 Maximum Entropy Method

3.1 Feature Function

We define binary-valued indicator function $f : X \times Y \rightarrow \{0, 1\}$ which divide $X \times Y$ into two subsets. This is called *feature function*, which expresses statistical properties of a language model.

The expected value of f with respected to $\tilde{p}(x, y)$ is defined such as:

$$\tilde{p}(f) = \sum_{x,y} \tilde{p}(x, y) f(x, y) \tag{3}$$

Thus training data are summarized as the expected value of feature function f.

The expected value of a feature function f with respected to $p(y|x)$ which we would like to estimate is defined such as:

$$p(f) = \sum_{x,y} \tilde{p}(x) p(y|x) f(x, y) \tag{4}$$

where $\tilde{p}(x)$ is the empirical probability distribution on X. Then, the model which we would like to estimate is under constraint to satisfy an equation such as:

$$p(f) = \tilde{p}(f) \tag{5}$$

This is called *the constraint equation.*

3.2 Maximum Entropy Principle

When there are feature functions $f_i (i \in \{1, 2, \ldots, n\})$ which are important to modeling processes, the distribution p we estimate should be included in a set of distributions defined such as:

$$\mathcal{C} = \{p \in \mathcal{P} \mid p(f_i) = \tilde{p}(f_i) \text{ for } i \in \{1, 2, \ldots, n\}\} \tag{6}$$

where $\mathcal{P}$ is a set of all possible distributions on $X \times Y$.

For the distribution p, there is no assumption except equation (6), so it is reasonable that the most uniform distribution is the most suitable for the training corpora. The conditional entropy defined in equation (7) is used as the mathematical measure of the uniformity of a conditional probability $p(y|x)$.

$$H(p) = -\sum_{x,y} \tilde{p}(x) p(y|x) \log p(y|x) \tag{7}$$

That is, the model $p*$ which maximizes the entropy H should be selected from $\mathcal{C}$.

$$p* = \underset{p \in \mathcal{C}}{\operatorname{argmax}} H(p) \tag{8}$$

This heuristic is called *the maximum entropy principle.*

3.3 Parameter Estimation

In simple cases, we can find the solution to the equation (8) analytically. Unfortunately, there is no analytical solution in general cases, and we need a numerical algorithm to find the solution.

By applying the Lagrange multiplier to equation (7), we can introduce the parametric form of p.

$$p_\lambda(y|x) = \frac{1}{Z_\lambda(x)} \exp\left(\sum_i \lambda_i f_i(x, y)\right) \tag{9}$$

$$Z_\lambda(x) = \sum_y \exp\left(\sum_i \lambda_i f_i(x, y)\right)$$

where each λ_i is the parameter for the feature f_i. p_λ is known as *Gibbs distribution.* Then, to solve $p* \in \mathcal{C}$ in equation (8) is equivalent to solve $\lambda*$ that maximize the log-likelihood:

$$\Psi(\lambda) = -\sum_x \tilde{p}(x) \log Z_\lambda(x) + \sum_i \lambda_i \tilde{p}(f_i) \tag{10}$$

$$\lambda* = \underset{\lambda}{\operatorname{argmax}} \Psi(\lambda)$$

Such $\lambda*$ can be solved by one of the numerical algorithm called *the Improved Iterative Scaling Algorithm* (Berger et al., 1996).

1. Start with $\lambda_i = 0$ for all $i \in \{1, 2, \ldots, n\}$
2. Do for each $i \in \{1, 2, \ldots, n\}$:

(a) Let $\Delta\lambda_i$ be the solution to

$$\sum_{x,y} \tilde{p}(x)p(y|x)f_i(x,y)\exp\left(\Delta\lambda_i f^{\#}(x,y)\right) = \tilde{p}(f_i) \quad (11)$$

where $f^{\#}(x,y) = \sum_{i=1}^{n} f_i(x,y)$

(b) Update the value of λ_i according to:

$$\lambda_i \leftarrow \lambda_i + \Delta\lambda_i$$

3. Go to step 2 if not all the λ_i have converged

To solve $\Delta\lambda_i$ in the step (2a), the Newton's method is applied to equation (11).

3.4 Feature Selection

In general cases, there exist a large collection $\mathcal{F}$ of candidate features, and because of the limit of machine resources, we cannot expect to obtain all $\tilde{p}(f)$ estimated in real-life. However, the Maximum Entropy Principle does not explicitly state how to select those particular constraints. We build a subset $\mathcal{S} \subset \mathcal{F}$ incrementally by iterating to adjoin a feature $\hat{f} \in \mathcal{F}$ which maximizes log-likelihood of the model to $\mathcal{S}$. This algorithm is called *the Basic Feature Selection* (Berger et al., 1996).

1. Start with $\mathcal{S} = \emptyset$
2. Do for each candidate feature $f \in \mathcal{F}$: Compute the model $p_{\mathcal{S}\cup f}$ using Improve Iterative Scaling Algorithm and the gain in the log-likelihood from adding this feature
3. Check the termination condition
4. Select the feature $\hat{f}$ with maximal gain
5. Adjoin $\hat{f}$ to $\mathcal{S}$
6. Compute $p_{\mathcal{S}}$ using Improve Iterative Algorithm
7. Go to Step 2

4 Maximum Entropy Model Learning of the Translation Rules

The art of modeling with the maximum entropy method is to define an informative set of computationally feasible feature functions. In this section, we define two models of feature functions for learning the translation rules.

Model 1: Co-occurrence Information

The first model is defined with co-occurrence information between words appeared in the corpus X.

$$f_w(x,y) = \begin{cases} 1 & (x \in W(d,w)) \\ 0 & (\text{otherwise}) \end{cases} \quad (12)$$

where $W(d,w)$ is a set of words which appeared within d words from $w \in X$ (in our experiments, $d = 5$). $f_w(x,y)$ expresses the information on w for predicting that x is translated into y (Figure 1).

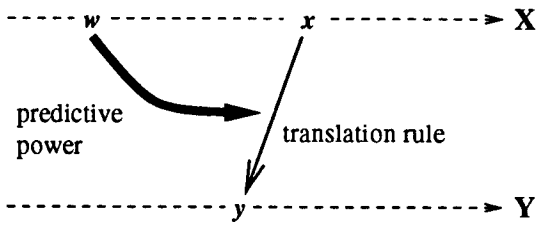

Figure 1: co-occurance information

Model 2: Morphological Information

The second model is defined with morphological information such as part-of-speech.

$$f_{t,s}(x,y) = \begin{cases} 1 & \begin{pmatrix} POS(x) = t \\ \text{and} \\ POS(y) = s \end{pmatrix} \\ 0 & (\text{otherwise}) \end{cases} \quad (13)$$

where $POS(x)$ is a part-of-speech tag for x. $f_{t,u}(x,y)$ expresses the information on part-of-speech t, s for predicting that x is translated into y (Figure 2). If part-of-speech tag-

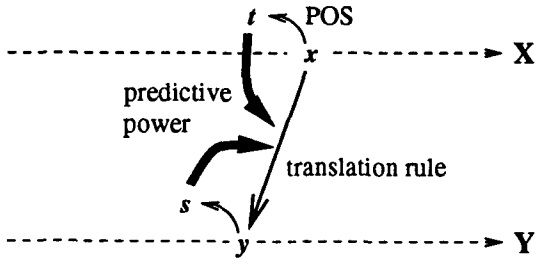

Figure 2: morphological information

gers for each language work extremely accurate, then these feature functions can be generated automatically.

5 Implementation

Computational cost associated with the model described above is too expensive to realize the system for learning the translation rules. We propose several methods to suppress the overhead.

An estimated probability $p_\lambda(y|x)$ for a pair of $(x, y) \in X \times Y$ which has not been observed as the sample data in the parallel corpora X, Y should be kept lower. According to equation (9), we can allow to let $f_i(x, y) = 0$ (for all $i \in \{1, \ldots, n\}$) for non-observed (x, y). Therefore, we will accept observed (x, y) only instead of all possible (x, y) in summation in equation (11), so that $p_\lambda(y|x)$ can be calculated much more efficiently.

Suppose that a set of (x, y) such that each member activates a feature function f is defined by:

$$D(f) = \{(x, y) \in X \times Y \mid f(x, y) = 1\} \quad (14)$$

Shirai et al. (1996) showed that if $D(f_i)$ and $D(f_j)$ were exclusive to each other, that is $D(f_i) \cap D(f_j) = \emptyset$, then λ_i and λ_j could be estimated independently. Therefore, we can split a set of candidate feature functions $\mathcal{F}$ into several exclusive subsets, and calculate $p_\lambda(y|x)$ more efficiently by estimating on each subset independently.

6 Experiments and Results

As the training corpora, we used 6,057 pairs of sentences included in Kodansya Japanese–English Dictionary, a machine-readable dictionary made by the Electrotechnical Laboratory. By applying morphological analysis for the corpora, each word was transformed to the infinitive form. We excluded words which appeared below 3 times or over 1,000 times from the target of learning. Consequently, our target for the experiments included 1,375 English words and 1,195 Japanese words, and we prepared 1,375 feature functions for model 1 and 2,744 for model 2 (56 part-of-speech for English and 49 part-of-speech for Japanese).

We tried to learn the translation rules from English to Japanese. We had two experiments: one of model 1 as the set of feature functions, and one of model 1 + 2. For each experiment, 500 feature functions were selected according to the feature selection algorithm described in section 3.4, and we calculated $p(y|x)$ in equation (9), that is, the probability that English word x is translated into Japanese word y. For each English word, all Japanese word were ordered by estimated probability $p(y|x)$, and we evaluated the recall rates by comparing the dictionary. Table 1 shows the recall rates for each experiment. The numbers for $\tilde{p}(x, y)$ are the recall rates when the empirical probability defined by equation (1) was used instead of the estimated probability. It is showed that the model 1 + 2 attains higher recall rates than the model 1 and $\tilde{p}(x, y)$.

Table 1: recall rates

	1st	~ 3rd	~ 10th
$\tilde{p}(x, y)$	44.55%	53.47%	58.42%
model 1	41.58%	63.37%	76.24%
model 1 + 2	58.29%	69.54%	80.13%

Figure 3 shows the log-likelihood for each model plotted by the number of feature functions in the feature selection algorithm. Notice that the log-likelihood for the model 1+2 is always higher than the model 1.

Thus, the model 1 + 2 is more effective than the model 1 for learning the translation rules.

However, the result shows that the recall

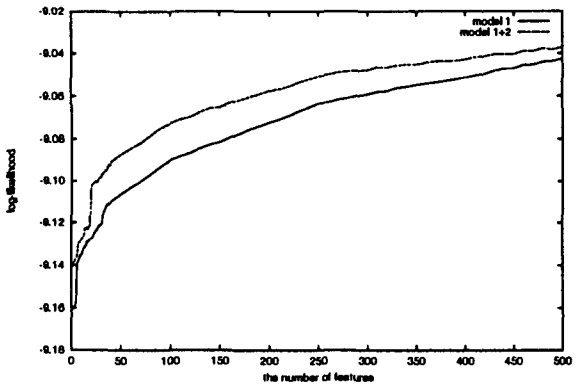

Figure 3: log-likelihood

rates of the '1st' for all models are not favorable. We consider that it is the reason for this to assume word-to-word translation rules implicitly.

7 Conclusions

We have described an approach to learn the translation rules from parallel corpora based on the maximum entropy method. As feature functions, we have defined two models, one with co-occurrence information and the other with morphological information. As computational cost associated with this method is too expensive, we have proposed several methods to suppress the overhead in order to realize the system. We had experiments for each model of features, and the result showed the effectiveness of this method, especially for the model of features with co-occurrence and morphological information.

Acknowledgments

We would like to thank the Electrotechnical Laboratory for giving us the machine-readable dictionary which was used as the training data.

References

L. E. Baum. 1972. An inequality and associated maximumization technique in statistical estimation of probabilistic functions of a markov process. *Inequalities*, 3:1–8.

Adam L. Berger, Stephen A. Della Pietra, and Vincent J. Della Pietra. 1996. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71.

Hiroyuki Kaji and Toshiko Aizono. 1996. Extracting word correspondences from bilingual corpora based on word co-occurrence information. In *Proceedings of the 16th International Conference on Computational Linguistics*, pages 23–28.

M. Kay and M. Röschesen. 1993. Text translation alignment. *Computational Linguistics*, 19(1):121–142.

J. D. Lafferty. 1993. A derivation of the inside-outside algorithm from the EM algorithm. *IBM Research Report*. IBM T.J. Watson Research Center.

Stephen Della Pietra, Vincent Della Pietra, and John Lafferty. 1995. Inducing features of random fields. Technical Report CMU-CS-95-144, Carnegie Mellon University, May.

Adwait Ratnaparkhi. 1997. A linear observed time statistical parser based on maximum entropy models. In *Proceedings of Second Conference On Empirical Methods in Natural Language Processing.*

Jeffrey C. Reynar and Adwait Ratnaparkhi. 1997. A maximum entropy approach to identifying sentence boundaries. In *Proceedings of the 5th Applied Natural Language Processing Conference.*

Ronald Rosenfeld. 1996. A maximum entropy approach to adaptive statistical language modeling. *Computer, Speech and Language*, (10):187–228.

Kiyoaki Shirai, Kentaro Inui, Takenobu Tokunaga, and Hozumi Tanaka. 1996. A maximum entropy model for estimating lexical bigrams (in Japanese). In *SIG Notes of the Information Processing Society of Japan*, number 96–NL–116.

Takehito Utsuro, Takashi Miyata, and Yuji Matsumoto. 1997. Maximum entropy model learning of subcategorizatoin preference. In *Proceedings of the 5th Workshop on Very Large Corpora*, pages 246–260, August.