数理解析研究所講究録
第1629巻 2009年 87-93

Integer Programming for a Phrase Alignment Problem on Statistical Machine Translation

Mikio Yamamoto*, Shunji Umetani+, Mitsuru Koshikawa* and Tomomi Matsui++
*University of Tsukuba, +Osaka University, ++Chuo University

1 Introduction

Realization of "Machine Translation (MT)" systems with humanlike capability has been the biggest dream in the research field of natural language processing. In 1947 Warren Weaver pointed out we can regard a decoder for decryption as a machine translation system, if we consider a foreign language as a code language (Weaver 1955). Over the past 60 years, though much many researchers and engineers in this field have tried building such a MT system, the complete one is still a dream. However, recently statistical methods have innovated the technological framework of MT, in which all translation rules are automatically estimated from vast amount of parallel text data (e.g. a collection of English-Japanese translated sentences pairs) using recent high performance computing powers.

In the framework of "Statistical Machine Translation (SMT)" two problems should be solved. The "modeling problem" is how to model and estimate a valid probability function of output candidate sentences conditioned by an input sentences. The "decoding problem" is how to find the best output sentence which maximizes the estimated probability function. The system to find the best output is called 'decoder' after Weaver. The search space of a decoder is extremely large, so we have to approximate the probability function and maximization process at multiple-stages.

In this paper, we will discuss an integer programming approach to solve a part of the decoding problem: the phrase alignment problem.

2 Phrase-Based Statistical Machine Translation

If we have a probabilistic distribution, $P(\boldsymbol{e}|\boldsymbol{f})$, of a sentences $\boldsymbol{e}$ in target (or output) language (e.g. English) given a input sentence $\boldsymbol{f}$ in source (or input) language (e.g. French), we can make a chance of translation error minimizing to find the output sentence $\hat{\boldsymbol{e}}$ which maximizes the probability of $P(\hat{\boldsymbol{e}}|\boldsymbol{f})$. This approach is called the "noisy channel model."

$$\begin{aligned}\hat{\boldsymbol{e}} &= arg\max_{\boldsymbol{e}} P(\boldsymbol{e}|\boldsymbol{f}) \\ &= arg\max_{\boldsymbol{e}} \frac{P(\boldsymbol{e})P(\boldsymbol{f}|\boldsymbol{e})}{P(\boldsymbol{f})} \\ &= arg\max_{\boldsymbol{e}} P(\boldsymbol{e})P(\boldsymbol{f}|\boldsymbol{e})\end{aligned}$$

$P(\boldsymbol{e})$ and $P(\boldsymbol{f}|\boldsymbol{e})$ are called a *language model* and a *translation model*, respectively. Note that the direction of the translation model is inverse (i.e. probabilities of source sentences given a target sentence). Language models are often approximated by the Markov models. A translation model is decomposed into local translation models based on bilingual dictionaries. There are many variations of the decomposition. Brown et al. (1993) proposed a basic model based on word-to-word alignment style in their paper, which is the origin of all recent SMT researches. However, in the past decade, the phrase-to-phrase alignment style has predominated, because its performance is much better than word-to-word based models and it can capsulate complex local translation process in phrases into a bilingual phrase dictionary (Koehn, Och and Marcu 2003).

In phrase-based models, the translation model is decomposed into probabilities of phrase "segmentations" of source and target sentences ($\bar{f}$ and $\bar{e}$, respectively), an probability of an "alignment" (a) between segmented phrases in target and source sentences, and phrase translation probabilities conditioned by the segmentations and alignment. Then it is marginalized.

$$\begin{aligned} P(f|e) &= \sum_{\bar{f}=f,\bar{e},a} P(f,\bar{f},\bar{e},a|e) \\ &= \sum_{\bar{f}=f,\bar{e}=e,a} P(\bar{e}|e)P(a|e,\bar{e})P(\bar{f}|e,\bar{e},a)P(f|\bar{f},e,\bar{e},a) \end{aligned}$$

Phrase segmentations $\bar{f}$ and $\bar{e}$ have complete information of the original source and target sentences f and e.

$$= \sum_{\bar{f},\bar{e},a} P(\bar{e}|e)P(a|\bar{e})P(\bar{f}|\bar{e},a)$$

In the above model, a target sentence e is segmented into a sequence of I phrases $\bar{e}_1^I (= e)$. An alignment a is a sequence of a_i (that is, $a = a_1^I$) which represents a position of a source phrase translated from a target phrase $\bar{e}_i$. That is, each phrase $\bar{e}_i$ in $\bar{e}_1^I$ is translated into a phrase $\bar{f}_{a_i}$.

If we assume that the segmentation probabilities are uniform , the translation probability can be decomposed into

$$\begin{aligned} P(f|e) &\propto \sum_{\bar{f},\bar{e},a} P(a|\bar{e})P(\bar{f}|\bar{e},a) \\ &\propto \sum_{\bar{f},\bar{e},a} P(a|\bar{e}) \prod_{i=1}^{I} \phi(\bar{f}_{a_i}|\bar{e}_i). \end{aligned}$$

Where $\phi(\bar{f}_{a_i}|\bar{e}_i)$ is a translation probability of aligned phrases in target and source sentences.

$P(a|\bar{e})$ is a reordering model which gives probabilities about position moving of phrases in a source sentence. We can decompose a reordering model into

$$\begin{aligned} P(a|\bar{e}) &= \prod_{i=1}^{I} P(a_i|\bar{e}, a_1^{i-1}) \\ &\approx \prod_{i=1}^{I} P(a_i|\bar{e}_i, a_{i-1}). \end{aligned}$$

We call this model a "lexicalized reordering model" because a decomposed probability is conditioned by the actual phrases $\bar{e}_i$'s (Koehn et al. 2005). Avoiding the problem of sparse training data, we only consider three reordering types: monotone order (m), swap with previous phrase (s), or discontinuous (d). We define a function $type(j,k)$ as the following.

$$type(j,k) = \begin{cases} m & \text{if } j = k-1 \\ s & \text{if } j-1 = k \\ d & \text{others.} \end{cases}$$

Using this function we get the final version of the reordering model.

$$P(a|\bar{e}) \approx \prod_{i=1}^{I} P(type(a_{i-1}, a_i)|\bar{e}_i).$$

3 Phrase Alignment Problems

Since the search space to compute the best target sentence in the phrase-based SMT model described in the previous section is large, the decoding problem is approximated as the following.

$$\begin{aligned}\hat{e} &= arg\max_{e} P(e)P(f|e) \\ &= arg\max_{e} P(e) \sum_{\bar{f}=f,\bar{e}=e,a} P(a|\bar{e})P(\bar{f}|\bar{e},a) \\ &\approx arg\max_{e} P(e) \max_{\bar{f}=f,\bar{e}=e,a} P(a|\bar{e})P(\bar{f}|\bar{e},a)\end{aligned}$$

Inside maximization for $\bar{f}$,$\bar{e}$ and a in the last formula is called a "phrase alignment problem." This approximation of summing by maximization is justified by the intuitive fact that probability mass of only a few correct segmentation and alignment is predominantly large and the other probability can be ignored. However, although the search space was dramatically reduced by this approximation, it remains too large to get the exact result. A real decoder uses a heuristic search algorithm and finds out a pseudo best result from the very limited space over the four variables of e, $\bar{f}$, $\bar{e}$ and a. In the next section, we will formulate the phrase alignment problem as an integer linear programming to develop the algorithm to compute the exact best result for three variables of $\bar{f}$, $\bar{e}$ and a, but e.

In the remains of this section, we define the realistic phrase alignment problem, because the current model of SMT becomes a little bit more complicated. In real SMT systems, the noisy channel model is extended to integrate the other information as effective for translation quality as possible using the log-linear model. This approach is called the "discriminative model." For example, in the noisy channel model we used only $P(f|e)$ as the (inverse) translation model. But it is known that the original translation model $P(e|f)$ is also effective for improving translation quality. For another example, we used $P(type(a_{i-1}, a_i)|\bar{e}_i)$ to compute a reordering probability. But we can improve translation quality to use richer conditioned probability $P(type(a_{i-1}, a_i)|\bar{e}_i, \bar{f}_{a_i})$ and reverse directional probability $P(type(a_i, a_{i+1})|\bar{e}_i, \bar{f}_{a_i})$. We can integrate the basic model and such additional probabilities or features into the log-linear model.

$$logP(e|f) = C + \sum_i w_i f_i(e, f)$$

Where $f_i(e, f)$ is the i-th probability or feature, w_i is the weight for it and C is a normalization constant which can be ignored in the maximization problem. The weights are determined by the discriminative training method. In the case of SMT, we commonly use a minimum error rate training (MERT) (Och and Ney 2003) which adjusts weights to maximize an automatic evaluation measure for translation quality such as BLEU (Papineni et al. 2002).

An example of the realistic phrase alignment problem is the following.

$$\begin{aligned}(\hat{\bar{f}}, \hat{\bar{e}}, \hat{a}) &= arg\max_{\bar{f},\bar{e},a} P(a|\bar{e})P(\bar{f}|\bar{e},a) \\ &= arg\max_{\bar{f},\bar{e},a} w_{p1} \prod_{i=1}^{I} \phi(\bar{f}_{a_i}|\bar{e}_i) \times w_{p2} \prod_{i=1}^{I} \phi(\bar{e}_i|\bar{f}_{a_i}) \\ &\qquad \times w_{r1} \prod_{i=1}^{I} P(type(a_{i-1}, a_i)|\bar{e}_i, \bar{f}_i) \times w_{r2} \prod_{i=1}^{I} P(type(a_i, a_{i+1})|\bar{e}_i, \bar{f}_i)\end{aligned}$$

All parameters of weights and values of probability functions are given. Fortunately, we can compile all given parameters for each phrase pair into two kinds of integrated parameters p and d indexed by an entry for a phrase pair in the dictionary. The integrated parameters p is determined by just a phrase pair and d is determined by a phrase pair and a reordering type.

When a phrase alignment system is given a sentence pair f and e, at the first it makes up a table of candidates of phrase pairs to match partly to f and e looking up in the dictionary. Then, from the

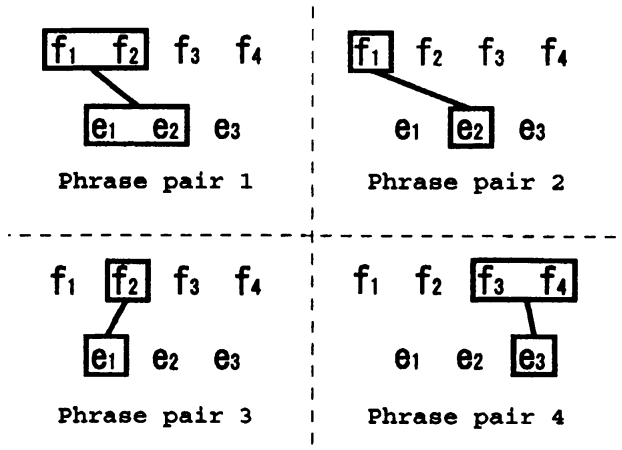

Figure 1: Candidates of phrase pairs for the input sentence pair.

table it computes a set of phrase pairs which covers all words in the sentence pair and maximizes the object function. Note that a output set of phrase pairs from the table determines the phrase alignment at the same time. To make a problem simple, we consider that the table of candidates of phrase pairs is also given as the part of the problem. The solution of the problem is a set of selected phrase pairs from the table.

The table of candidates of phrase pairs for the input sentence pair is defined as the following. We assume that there are four phrase pairs shown in Figure 1 as candidates for the input sentence pair $\boldsymbol{f} = f_1, f_2, f_3, f_4$ and $\boldsymbol{e} = e_1, e_2, e_3$ where f_i and e_i is words in source and target language, respectively. In the Figure 1, boxes denote phrases in a input sentence and lines denote alignments between phrases in source and target sentences. The table of candidate phrase pairs is represented by two matrices F and E. The i-th column vectors in F and E denote word sequences of phrases of the i-th phrase pair, and words in the phrase are expressed with 1's and words out of the phrase with 0's. For example, the next F and E correspond to four candidates of the phrase pairs in Figure 1.

$$F = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}, E = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

In the next section, we formulate phrase alignment problems as an integer linear programming when the table and parameters are given.

4 Phrase Alignment as an integer programming problem

4.1 Simple Phrase Alignment Problems

In this subsection, we consider a simple phrase alignment problem, in which we assume the reordering probability is uniform, that is the reordering model is ignored. At the first, we introduce binary variables $x_k \in \{0, 1\}$ which represent whether the k-th phrase pair in the table selected (1) or not (0) as a member of an output set of phrase pairs. DeNero and Klein (2008) formulate this simple version

of the phrase alignment problem as the following.

$$\left|\begin{array}{ll} \text{maximize} & \sum_{k=1}^{K} x_k p_k \\ \\ \text{subject to} & \boldsymbol{F}\boldsymbol{x} = \mathbf{1}, \\ & \boldsymbol{E}\boldsymbol{x} = \mathbf{1}, \\ & x_k \in \{0,1\} \quad (1 \leq k \leq K). \end{array}\right.$$

Where p_k is a compiled parameter of the k-th phrase pair, K is the number of candidates of phrase pairs and $\mathbf{1}$ is $(1 \cdots 1)^T$.

4.2 Full Phrase Alignment Problems

It's difficult to extend the formulation in the previous subsection to one of incorporating the reordering models, because auxiliary variables x_k have no information of position relationship between phrase pairs. So we introduce a graph representation of the table of phrase pairs in the target side instead of E and the second auxiliary variables y_a.

Figure 2 shows an example of the graph representation of the candidate phrases in the target side sentence. Boxes denote phrases and lines denote connections between adjacent phrases. Using the graph representation, we incorporate the compiled parameters for the reordering models as weights d_a on the connections. The feasible sets of phrases are expressed with paths starting from the node s to the node g. We can clearly ovserve two feasible paths on the graph in Figure 2.

New auxiliary binary variables y_a means whether the a-th connection is on the feasible path ($y_a = 1$) or not ($y_a = 0$). Using new auxiliary variables y_a, we can formulate a full phrase alignment problem as the following.

$$\left|\begin{array}{ll} \text{maximize} & \sum_{k=1}^{K} x_k p_k + \sum_{a=1}^{A} y_a d_a \\ \\ \text{subject to} & \boldsymbol{F}\boldsymbol{x} = \mathbf{1}, \\ & \boldsymbol{M}\boldsymbol{y} = \boldsymbol{b}, \\ & \boldsymbol{N}\boldsymbol{y} = \boldsymbol{x}, \\ & x_k \in \{0,1\} \quad (1 \leq k \leq K), \\ & y_a \in \{0,1\} \quad (1 \leq a \leq A). \end{array}\right.$$

Where the parameters d_a in the object function denotes the compiled parameters for the reordering probabilities and its weights. We can regard d_a as a weight on the a-th connection in the graph representation. The equation $\boldsymbol{M}\boldsymbol{y} = \boldsymbol{b}$ represents the "conservation law of flow," which is the standard

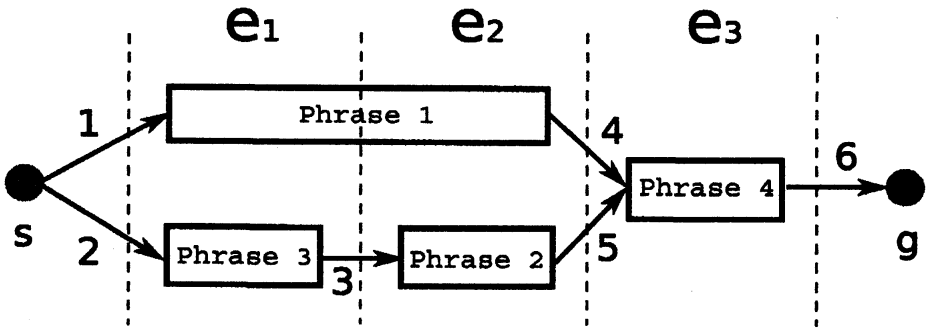

Figure 2: An example of the graph representation of the target side phrase candidates in Figure 1.

technique for guaranteeing valid paths. The equation $Ny = x$ represents the relationship between x and y. If the connection variable y_a is 1, x_k of both side of the connection must be 1. The symbol A denotes the number of connections of the phrase graph in the target side.

For example, the "conservation law of flow" for Figure 2 is the following.

$$\begin{pmatrix} -1 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

The first item is M and each column corresponds to the nodes (phrase) s, phrase1, ...phrase4 and g. The fifth line vector of M represents the conservation law for the fourth phrase (node); $y_4 + y_5 = y_6$. An example of the equation $Ny = x$ for Figure 2 is the following.

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix}$$

5 Experiments and Conclusion

We build up a dictionary of phrase pairs the number of entries of wihch is 60 million from 2 million parallel Japanese-English sentence pairs of the training data at the NTCIR-7 (Fujii et al. 2008) using the script within Moses package (Koehn et al. 2007). We used CPLEX version 11.0 as the solver for the integer programming and solved the full phrase alignment problem for a few hundred thousand sentence pairs for the test. Figure 3 shows an example of a phrase alignment for a real Japanese-English sentence pair computed by CPLEX. In spite of such realistic setting and data, average time to compute the best alignment for one sentence pair was about a few hundred milliseconds. We plan to apply the method in this paper to the reranking problem in order to improve the quality of statistical machine translation.

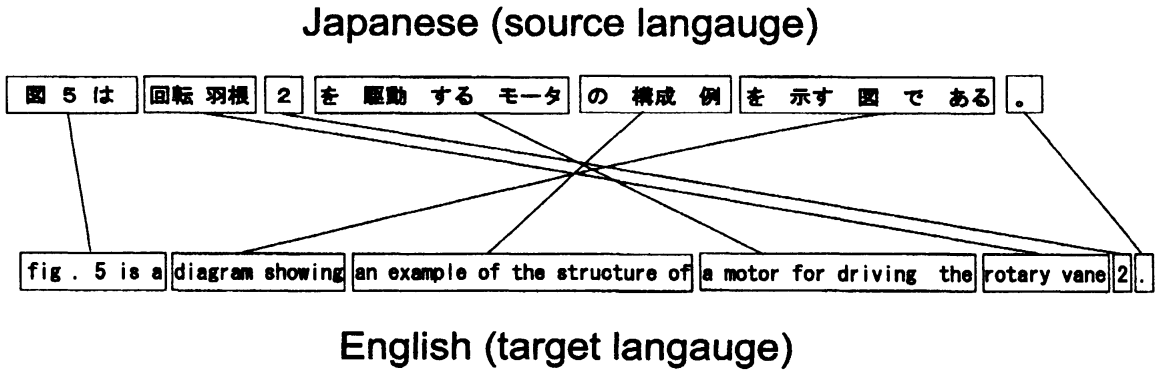

Figure 3: An example of a phrase alignment result for a Japanese-English sentence pair.

References

Brown, P. F., S. A. Della Pietra, V. J. Della Pietra and R. L. Mercer. 1993. The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*, Vol.19, No.2, pages 263-311.

DeNero, J. and D. Klein. 2008. The complexity of phrase alignment problems. In Proc. of ACL-08, pages 25–28.

Fujii, A., M. Yamamoto, M. Utiyama and T. Utsuro. 2008. Overview of the patent translation task at the NTCIR-7 workshop. In Proc. of NTCIR-7, pages 389–400.

Koehn, P., A. Axelrod, A. B. Mayne, C. Callison-Burch, M. Osborne and D. Talbot. 2005. Edinburgh system description for the 2005 IWSLT speech translation evaluation. In Proc. of IWSLT-2005.

Koehn, P., F. J. Och and D. Marcu. 2003. Statistical phrase-based translation. In Proc. of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics, pages 48–54.

Koehn, P. et al. 2007. Moses: open source toolkit for statistical machine translation. In Proc. of ACL-07, demonstration session.

Och, F. J. and H. Ney. 2003. Minimum error rate training in statistical machine translation. In Proc. of ACL-03, pages 160–167.

Papineni, K., S. Roukos, T. Ward and W. Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In Proc. of ACL-02, pages 311-318.

Weaver, Warren. 1955. Translation. *Machine Translation of Languages: Fourteen Essays*, W.N.Locke and A.D. Booth (eds.). (Reprinted in *Readings in Machine Translation*, S.Nirenburg, H.Somers and Y.Wilks (eds.), MIT Press, 2003)