INTEGRATED PHRASE SEGMENTATION AND ALIGNMENT ALGORITHM FOR STATISTICAL MACHINE TRANSLATION

Ying Zhang Stephan Vogel Alex Waibel

Language Technologies Institute, School of Computer Sciences, Carnegie Mellon University
5000 Forbes Ave. Pittsburgh, PA 15213-3891 U.S.A.
{joy+,vogel+,ahw+}@cs.cmu.edu

ABSTRACT

We present an integrated phrase segmentation/alignment algorithm (ISA) for Statistical Machine Translation. Without the need of building an initial word-to-word alignment or initially segmenting the monolingual text into phrases as other methods do, this algorithm segments the sentences into phrases and finds their alignments simultaneously. For each sentence pair, ISA builds a two-dimensional matrix to represent a sentence pair where the value of each cell corresponds to the Point-wise Mutual Information (MI) between the source and target words. Based on the similarities of MI values among cells, we identify the aligned phrase pairs. Once all the phrase pairs are found, we know both how to segment one sentence into phrases and also the alignments between the source and target sentences. We use monolingual bigram language models to estimate the joint probabilities of the identified phrase pairs. The joint probabilities are then normalized to conditional probabilities, which are used by the decoder. Despite its simplicity, this approach yields phrase-to-phrase translations with significant higher precisions than our baseline system where phrase translations are extracted from the HMM word alignment. When we combine the phrase-to-phrase translations generated by this algorithm with the baseline system, the improvement on translation quality is even larger.

Keywords: Phrase Alignment, Phrase Segmentation, Statistical Machine Translation

1. INTRODUCTION

Building the sub-sentential alignment of a bilingual corpus and extracting the bilingual lexicon from it is of crucial importance in Statistical Machine Translation (SMT) systems. Early works, such as [Brown 97] presented methods of extracting bilingual lexicons of words relying on the distribution of the words in the set of parallel sentences. [Melamed 97] used a fast and greedy algorithm called Competitive Linking to find the word-to-word translation equivalences. This approach is based on the assumption that words are translated one-to-one. The assumption helps to avoid the indirect associations. Similar work is presented in [Gaussier 98].

As a reasonable step beyond the word-to-word alignment model, several researchers proposed using phrase-to-phrase alignment models. The advantage is that word context and local reordering are implicitly taken into account in phrase-to-phrase alignment models [Och et al. 2000]. The alignment template approach by [Och et al. 99] is based on a word-aligned training corpus and makes use of single word-based alignment models. [Marcu 2001] trained IBM translation model 4 on the Hansard corpus and used the Viterbi alignment of each sentence to extract tuples of the form <*English phrase; French phrase; Alignment between the words*>. Two constraints were used in the extraction: (1) only "contiguous" alignments were selected and (2) the English and French phrases should contain at least two words.

Other research starts from the phrasal level segmentation. For example, [Zhang 2001] used the monolingual language model to identify phrases in the Chinese and English corpus and create segmentation. Alignments were built on the segmented data to extract phrase-to-phrase translations. Zhang reported positive results for an Example Based Machine Translation system and showed that this method compensates for the over-segmentation problem caused by the word level segmentation.

In the rest of this paper, we will describe our integrated phrase segmentation and alignment algorithm. This algorithm does not require an initial word-to-word alignment, nor does it require an initial segmentation on the monolingual text. It uses the Point-wise Mutual Information between the source and target words to identify the phrase pairs. Once all the phrase pairs are found, we know both how to segment one sentence into phrases and also the alignments between the source and target sentences. We use monolingual bigram language models to estimate the joint probabilities for identified phrase pairs. The joint probabilities are then normalized to conditional probabilities, which are used by the decoder.

2. INTEGRATED PHRASE SEGMENTATION AND ALIGNMENT ALGORITHM

2.1 Algorithm

Represent a sentence pair <*F, E*> as a two-dimensional bi-text map $D_{m \times n}$, where m is the number of words in *F*, n in *E*, respectively. We can segment F into c' phrases: $\tilde{f}_1, \tilde{f}_2, \ldots, \tilde{f}_c, \ldots, \tilde{f}_{c'}$, and the same to E as segmenting it into d' phrases: $\tilde{e}_1, \tilde{e}_2, \ldots, \tilde{e}_d, \ldots, \tilde{e}_{d'}$. We want to partition D into box-shaped regions, where each region corresponds to an aligned phrase pair. Figure 1 shows an example of such partition. The goal is to find a partition with the maximum value of the joint probabilities of phrase pairs, subject to the constraint that no word belongs to more than one phrase pairs.

We use greedy-search to find the partition. To measure the "goodness" of translating a source word to a target word, we use the value of *Point-wise Mutual Information (MI)* between these two words. The *MI* values for source/target word pairs are then used to identify the potential phrase pairs for each sentence pair. Among all the possible phrase pairs that can be found for one sentence pair, select those that yield maximum $\prod_{<\tilde{f}_c, \tilde{e}_d>} P(<\tilde{f}_c, \tilde{e}_d>)$.

Point-wise Mutual Information (MI) between word *e* and *f* is defined as:

$$I(e,f) = \log_2 \frac{P(e,f)}{P(e)P(f)} \qquad (2.1.1)$$

The higher the *I(e, f)*, the more likely *e* is associated with *f*, or in other words, *e* is more likely to be the translation of *f*, and vice versa.

A sentence pair <*F, E*> can then be represented as a two dimensional matrix, where the *X* axis from left to right are the target words and *Y* axis from top down are the source words. The value of each cell in the matrix is the *MI* value of the source/target word pairs accordingly. Figure 1 shows an example of the *MI* distribution, where the grayscale of a cell corresponds to its *MI* value.

We observe that if the translation for e_1e_2 is *f*, $I(e_1, f)$ should be very *"similar"* to $I(e_2, f)$. Based on this natural and reasonable observation, one can identify aligned phrase pairs based on the similarity of the *MI* value of word pairs.

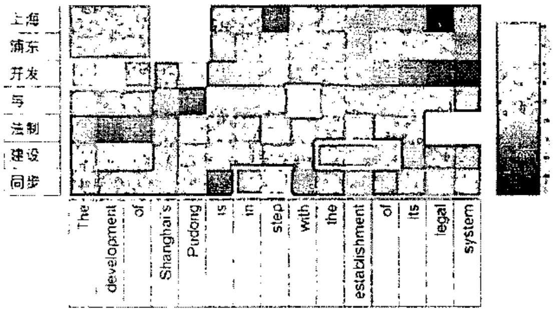

Figure 1. A sample sentence pair represented by a two-dimension matrix. Grayscale of *cell[x,y]* corresponds to *I(x,y)*

The algorithm is described as following:

Given a sentence pair *(F, E)*, construct a two-dimensional matrix $R_{m\times n}$. $R[i,j]=I(f_i, e_j)$. *F* can be represented as $f_1 f_2 \ldots f_i \ldots f_m$, where *m* is the number of words in *F*. *E* can be represented as $e_1 e_2 \ldots e_j \ldots e_n$, where n is the number of words in *E*. All the cells in *R* are initiated as *"free"* cells.

(1) Amongst all the *"free"* cells in the matrix, find the one with the highest *MI* value. $[i^*, j^*] = \arg\max_{i,j} R[i,j]$. Call this cell$[i^*, j^*]$ the current *"seed"* cell.

(2) Expand the *"seed"* cell to the largest possible rectangle regions $(r_{start}, r_{end}, c_{start}, c_{end})$ under two constraints:

   a. The *MI* value of all the cells *[i',j']* in the expanded region should be *"similar"* to $R[i^*, j^*]$, i.e. for all

   $$[i',j'],\ \frac{I(i',j')}{I(i^*,j^*)} \geq threshold$$

   b. Do not expand the region if it will block the *"free"* cells with higher *MI* value.

   This rectangle region represents a phrase pair.

(3) Mark all the free cells *[r, c]* in *R* where

   $$r_{start} \leq r \leq r_{end} \text{ or } c_{start} \leq c \leq c_{end} \text{ as}$$

   "blocked". They are not *"free"* anymore.

(4) If there are still any "free" cells, go to step (1). Otherwise, output all the phrase pairs found.

2.2 Example

To demonstrate how the ISA algorithm works, let's consider the following example:

Given a Chinese sentence *F*: *上海 浦东 开发 与 法制 建设 同步* and its translation *E*: *The development of Shanghai's Pudong is in step with the establishment of its legal system*, we first construct a two-dimensional matrix *R* for this sentence pair (Table 1). In this table $R[i,j]=I'(f_i, e_j)$. Here *I'* is a modified version of *Point-wise Mutual Information* where word context is encapsulated and negative values are possible. To keep things simple, we will still use the original *MI* concept in the rest of the paper.

上海	0	-0.50	0.5	0.4	7.2	6.2	-2.6	-0.8	-7.1	0.4	-1.4	-3.7	-3.3	-3.9	-10.0	-3.0
浦东	1	0.44	-0.4	0.6	4.8	8.4	-0.6	-3.0	-1.1	0.0	0.1	-3.1	-1.7	-0.9	-1.5	-3.6
开发	2	2.66	7.1	1.0	-1.4	2.8	-4.7	-4.0	-2.3	-0.2	-0.7	-2.6	-3.8	-4.8	-6.7	-7.2
与	3	-0.05	2.0	-0.5	-4.8	-7.0	1.2	-0.1	0.4	4.9	2.4	-0.2	0.1	0.6	1.6	-2.0
法制	4	-3.66	-5.8	-5.2	-3.0	1.3	-0.9	-2.5	2.6	-1.1	0.3	3.6	-1.4	-0.8	5.8	5.6
建设	5	-1.71	1.5	1.5	-3.5	-1.3	0.7	-0.2	-2.2	-3.0	1.8	2.7	4.1	-3.3	-0.4	-3.0
同步	6	-2.44	-3.8	-3.4	-3.5	-2.1	-6.4	0.4	3.8	-5.2	-1.5	3.1	-3.2	0.6	3.3	0.8
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
		The	development	of	Shanghai's	Pudong	is	in	step	with	the	establishment	of	its	legal	system

Table 1. MI values for the sample sentence pair

At the starting point, all the cells are *"free"*. Cell[1,4] *(浦东, Pudong)* has the highest *MI* value: 8.4. Mark it as the *"seed"* cell.

Since the neighboring cells: cell[0,3] *(上海, Shanghai's) I*=7.2, cell[0,4] *(上海, Pudong's) I*=6.2 and cell[1,3] *(浦东, Pudong) I*= 4.8, all have MI values *"similar"* to 8.4, we expand the seed cell to the rectangle regions (0,1,3,4). And now we have identified the first phrase pair as: *(上海 浦东, Shanghai's Pudong)*. After that, mark all the cells on row 0, row 1 and cells on column 3, column 4 as *"blocked"*.

Among the rest of the *"free"* cells, cell[2,1] *(开发, development)* has the highest MI value. We expand this cell to its neighbors and identified the second phrase pair: *(开发, the development of)*.

As the algorithm goes on, we will find the remaining phrase pairs for this sentence pair as: *(法制, legal system)*, *(建设, the establishment of)*, *(与, with)* and *(同步, in step)*.

3. ESTIMATE THE JOINT PROBABILITIES FOR PHRASE PAIRS

Once the sentence pair is segmented/aligned into phrase pairs, we need to estimate the joint probabilities for these phrase pairs. We will use these joint probabilities later to estimate the conditional translation probabilities, which are needed for the SMT decoder.

It is computationally hard to directly estimate $P(<\tilde{f}_c,\tilde{e}_d>)$ using the maximum likelihood estimation from the training data. Since we do not know the optimal segmentation during training, we cannot decide on the phrase boundaries. Even if we could do so, the data sparseness problem will become much worse at the phrase level than at the word level. In this paper, we estimate the joint probability for a phrase pair indirectly.

If we consider a phrase as one entity, the association between an English phrase $\tilde{e}_d$ and a foreign language $\tilde{f}_c$ phrase can be seen through the associations of their constituent words (Figure 2).

Thus, the joint probability of $\tilde{f}_c$ and $\tilde{e}_d$ can be estimated through the joint probabilities of English/Foreign word pairs.

$$\begin{aligned} P(\tilde{f}_c,\tilde{e}_d) &= \sum_{c\_i,d\_j} P(\tilde{f}_c,\tilde{e}_d,f_{c\_i},e_{d\_j}) \\ &= \sum_{c\_i,d\_j} P(\tilde{f}_c\,|\,f_{c\_i})P(f_{c\_i},e_{d\_j})P(\tilde{e}_d\,|\,e_{d\_j}) \end{aligned} \quad (3.1)$$

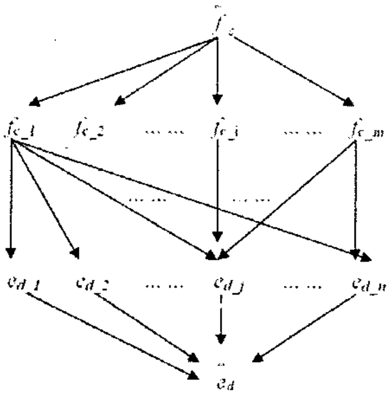

Figure 2. Estimate Joint Probability of Phrases

Under the independence assumption, we have:

$$P(\tilde{f}_c\,|\,f_{c\_i}) = \prod_{i'=1}^{m} P(f_{c\_i'}\,|\,f_{c\_i}) \quad (3.2)$$

And

$$P(\tilde{e}_d\,|\,e_{d\_j}) = \prod_{j'=1}^{n} P(e_{d\_j'}\,|\,e_{d\_j}) \quad (3.3)$$

The conditional probabilities $P(f_{c\_i'}\,|\,f_{c\_i})$ and $P(e_{d\_j'}\,|\,e_{d\_j})$ can be estimated from the training data or other monolingual data.

The output of the ISA algorithm is now a list of phrase-to-phrase translations, such as, *@PHRASE # 国家 杜马# state duma # 14.2779*, where 14.2779 is the cost (minus log of the joint probability) of this phrase pair. This list is then passed through a post processing script to convert the joint probabilities to conditional probabilities to fit into the SMT decoder.

4. EXPERIMENTS

[Och 2000], and [Ahrenberg 2000] similarly, proposed measuring the quality of an alignment *model using the quality of the Viterbi alignment* compared to a manually produced alignment (gold standard). Since our purpose of alignment is for translation, we evaluated our model by directly applying the phrase-to-phrase translations to the decoder and measuring the quality of the translations. Our experiments were based on the NIST Chinese-English Machine Translation evaluation package (NIST). We trained the system on the small data track of the June 2002 evaluation set and tested the system on the 2001 dry run testing data (Table 2).

	Sentences	**Chinese Words**	**English Words**
Training	3540 pairs	90K	115K
Testing	993	26K	

Table 2. Training and Testing Data

Both training and testing data were normalized using our data preprocessing scripts. The text normalization process includes: converting the Roman characters represented by Chinese encoding (two-bytes) to their one-byte ASCII equivalences; translating the Chinese number string to the Arabic number format and tag those in the training data as *<NUMBERS>*; converting all the uppercase English letters to lower case; inserting spaces around the punctuation marks. Chinese training and testing data were pre-segmented using the LR-segmenter [Zhang] with a word list provided by Language Data Consortium (LDC). This word frequency list contains about 44K word entries.

The decoder of our Statistical Machine Translation system works in two stages: First, the word-to-word translations and the phrase-to-phrase translations and, if available, other specific information, like named-entity translation tables are used to generate a translation lattice. A standard n-gram language model is then applied to find the best path in this lattice. Details of the decoder can be found in [Vogel et al. 2003].

For the baseline system, the translation model includes the LDC glossary, the word-to-word translations generated by IBM Model 1, and the *phrase-to-phrase translations extracted from the* HMM word alignment. The language model is built on the 20M-words 2001 Xinhua English news corpus.

To build the baseline, HMM word alignment [Vogel et al. 96] was first trained. The conditional probability of a source sentence given a target sentence is estimated as:

$$\begin{aligned} P(f \mid e) &= \sum_a P(f_1^J, a_1^J \mid e_1^I) \\ &= \sum_a \prod_j P(f_j, a_j \mid f_1^{j-1}, a_1^{j-1}, e_1^I) \\ &= \sum_a \prod_j P(a_j \mid a_{j-1}) P(f_j \mid e_{a(j)}) \qquad (4.1). \\ &\sim \max_a \prod_j P(a_j \mid a_{j-1}) P(f_j \mid e_{a(j)}) \end{aligned}$$

Over the Viterbi path found for each sentence pair, apply the following algorithm to extract the phrase-to-phrase translations up to L words long:

```
For each start position $j_1$ in source sentence
  For each length $l$ up to $L$
    $j_2 = j_1 + l - 1$
    Write source n-gram $j_1 \dots j_2$
    Find $min\_i = min\{a(j_1), \dots, a(j_2)\}$
    Find $max\_i = max\{a(j_1), \dots, a(j_2)\}$
    Write target n-gram $min\_i \dots max\_i$
```

To evaluate translation quality, we used the NIST MTeval scoring script [NIST]. There are two major components in the NIST metric: the *modified n-gram precision score(Prec)* and the *length penalty(Len)*. The former evaluates how "close" a hypothesis is as compared to the human generated translations, and the length penalty is used to penalize the short hypothesis to balance the precision scores.

The results of translation quality are listed in Table 3.

	Precision Score (Prec)	Length Penalty (Len)	FinalScore =Prec*Len
Baseline	6.7724	1.0000	6.7724
ISA	6.9664	0.9735	6.7818
ISA+ Baseline	7.0471	0.9975	7.0645

Table 3. NIST Score of Translation Quality

The phrases generated by ISA are on average shorter than the phrases extracted by HMM alignment. ISA has a higher precision score than the baseline, yet it has a higher length penalty. When the two approaches were combined, the precision and the length penalty were balanced and yielded a better translation score.

Student's t-test was applied at the sentence level (degree of freedom is 992) to calculate the statistical significance between the different systems. Results are show in Table 4 and Table 5.

Precision Scores	t value	Confidence Level
ISA vs. Baseline	6.6084	99.99%
ISA+Baseline vs. Baseline	9.0772	99.99%

Table 4. Student's t-test on Precision scores

Final Scores	t value	Confidence Level
ISA vs. Baseline	1.6890	95%
ISA+Baseline vs. Baseline	4.1007	99.99%

Table 5. Student's t-test on Final Scores

5. CONCLUSION

In this paper, we presented a simple and effective integrated segmentation/alignment model. Unlike [Marcu 2002], this algorithm does not require a first step of determining high-frequency n-grams in the bilingual corpus for phrase candidates. The purposes of phrase identification and alignment are achieved at the same time. We used the similarity of Point-wise Mutual Information to identify possible phrase pairs, and use the monolingual conditional probability to estimate the joint probabilities for phrase pairs.

The experiments showed that this approach yields better phrase-to-phrase translations than phrase alignments extracted from the HMM word alignments. The improvement is statistically significant when evaluated using the NIST automatic evaluation metric. And when combined with HMM phrase-to-phrase translations to balance the precision and the recall (length penalty), the improvement on translation quality is even higher.

6. ACKNOWLEDGEMENTS

We would like to express our appreciations to Jian Zhang, Guy Lebanon and Ashish Venugopal for their inspiring discussions and suggestions for this paper.

7. REFERENCES

[1] Lars Ahrenberg, Magnus Merkel, Anna Sågvall Hein & Jörg Tiedemann, "Evaluating Word Alignment Systems," *Proceedings of the Second International Conference on Linguistic Resources and Evaluation, LREC-2000*, vol. III, pp. 1255-1261. Athens, Greece, June 2000.

[2] Ralf D. Brown, "Automated Dictionary Extraction for 'Knowledge-Free" Example-Based Translation," *Proceedings of the Seventh International Conference on Theoretical and Methodological Issues in Machine Translation*, pp. 111-118. Santa Fe, July 23-25, 1997.

[3] Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer, "The Mathematics of Statistical Machine Translation: Parameter Estimation," *Computational Linguistics*, vol. 19, no. 2, pp. 263--311, 1993.

[4] Éric Gaussier, "Flow Network Models for Word Alignment and Terminology Extraction from Bilingual Corpora," *Proceedings of the 36th Annual Meeting of the association for*

Computational Linguistics and the 17th International Conference on Computational Linguistics. COLING-ACL'98, Montreal, Canada, August 1998.

[5] Daniel Marcu, "Towards a Unified Approach to Memory- and Statistical-Based Machine Translation," *Proceedings of ACL-2001*, Toulouse, France, July 2001.

[6] Daniel Marcu and William Wong, "A Phrase-Based, Joint Probability Model for Statistical Machine Translation," *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-2002)*, Philadelphia, PA, July 6-7, 2002.

[7] Dan Melamed, "A Word-to-Word Model of Translational Equivalence," *Proceedings of 35th Conference of the Association for Computational Linguistics (ACL'97),* Madrid, Spain, 1997.

[8] Dan Melamed, *Empirical Methods for Exploiting Parallel Text*, the MIT Press, 2001.

[9] NIST MT evaluation kit version 9. Available at: http://www.nist.gov/speech/tests/mt/.

[10] Franz Josef Och, Christoph Tillmann, Hermann Ney, "Improved Alignment Models for Statistical Machine Translation," *Proceedings of the Joint Conference of Empirical Methods in Natural Language Processing and Very Large Corpora*, pp. 20-28. University of Maryland, College Park, MD, June 1999.

[11] Franz Josef Och and Hermann Ney, "A Comparison of Alignment Models for Statistical Machine Translation," *Proceedings of the 18th International Conference on Computational Linguistics,* Saarbrucken, Germany, July 2000.

[12] Kishore Papineni et al, "Bleu: a Method for Automatic Evaluation of Machine Translation," *IBM Research Report*, RC22176, September 2001.

[13] Stephan Vogel, Hermann Ney, and Christoph Till-mann, "HMM-based Word Alignment in Statistical Translation," *Proceedings of COLING '96: The 16th International Conference on Computational Linguistics*, pp. 836-841. Copenhagen, August 1996.

[14] Stephan Vogel, Ying Zhang, Fei Huang, Alicia Tribble, Ashish Venogupal, Bing Zhao, Alex Waibel, "The CMU Statistical Translation System," to appear in the *Proceedings of MT Summit IX*, New Orleans, LA, U.S.A., September 2003.

[15] Ying Zhang. Bi-directional Maximum Word Frequency Chinese Word Segmenter. http://projectile.is.cs.cmu.edu/research/index.html#tools

[16] Ying Zhang, Ralf D. Brown, Robert E. Frederking and Alon Lavie, "Pre-processing of Bilingual Corpora for Mandarin-English EBMT," *Proceedings of MT Summit VIII*, Santiago de Compostela, Spain, September 2001.

[17] Ying Zhang, Ralf D. Brown, and Robert E. Frederking, "Adapting an Example-Based Translation System to Chinese," *Proceedings of Human Language Technology Conference 2001,* San Diego, CA, March 2001.