Example-based Machine Translation Pursuing Fully Structural NLP

Sadao Kurohashi, Toshiaki Nakazawa, Kauffmann Alexis, Daisuke Kawahara

Graduate School of Information Science and Technology
University of Tokyo
{kuro, nakazawa, alexis, kawahara}@kc.t.u-tokyo.ac.jp

Abstract

We are conducting Example-Based Machine Translation research aiming at the improvement both of structural NLP and machine translation. This paper describes UTokyo system challenged IWSLT05 Japanese-English translation tasks.

1. Introduction

We are conducting research on Example-Based Machine Translation, or EBMT [1] aiming at the improvement both of structural NLP and machine translation.

Machine translation has been actively studied recently, and the major approach is Statistical Machine Translation, or SMT. EBMT and SMT have something in common and something different. The important common feature is to use bilingual corpus, or translation examples, for the translation of new inputs. Both methods exploit translation knowledge implicitly embedded in translation examples, and make MT system maintenance and improvement much easier compared with Rule-Based Machine Translation.

The difference is that SMT supposes bilingual corpus is the only available resource (but not a bilingual lexicon and parsers); EBMT does not consider such a constraint. SMT basically combines words or phrases (relatively small pieces) with high probability [2]; EBMT tries to use larger translation examples. When EBMT tries to use larger examples, it had better handle examples which are discontinuous as a word-string, but continuous structurally. Accordingly, though it is not inevitable, EBMT naturally seeks syntactic information.

The difference in the problem setting is important. SMT is a natural approach when linguistic resources such as parsers and a bilingual lexicon are not available. On the other hand, in case of such linguistic resources are available, it is also natural to see how accurate MT can be achieved using all the available resources.

We chose the latter problem setting and conducting EBMT research, and here we would like to mention two reasons we chose this setting.

One reason is that we are aiming at the improvement of structural NLP. We have been conducting research on Japanese morphological analyzer, parser, and anaphora/omission analyses. MT is considered as an application of these fundamental technologies. Amelioration of fundamental NLP technologies naturally improves applications, and applications give some feedback to fundamental NLP, pointing the shortcomings. Needless to say, MT is not the only NLP application, and monolingual NLP applications such as man-machine interface and information retrieval can benefit from the improvement of fundamental NLP.

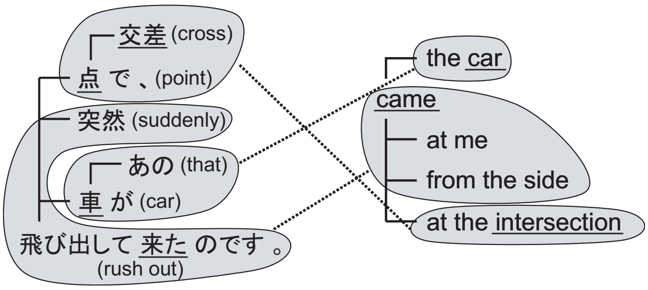

Figure 1: An example of parallel sentence alignment. (The root of a tree is placed at the extreme left and phrases are placed from top to bottom. Correspondences of underlined words were detected by a bilingual dictionary.)

The second point is that, in practice, we often encounter cases to which EBMT problem setting is suitable. That is, there is no huge bilingual corpus which enables SMT, but some very similar translation examples are available, and it would be nice if automatic translation or translation assistance can be provided by exploiting the examples. For example, translation of manuals when translations of the old version manuals are available, and patent translation when translations of the related patents are available. Or, in the translation of an article, the translations to a certain point can be used effectively as translation memory step by step, because the same or similar expressions/sentences are often used in an article. In such cases, EBMT approach is suitable which tries to find larger translation examples.

This paper describes our Japanese-English EBMT system, UTokyo, challenged to IWSLT05, and reports the evaluation results and discussion.

2. Alignment of Parallel Sentences

Our system consists of two modules: an alignment module for parallel sentences and a translation module retrieving appropriate translation examples and combining them. First, we explain the alignment module.

The alignment of Japanese-English parallel sentences is achieved by the following steps, using a Japanese parser, an English parser, and a bilingual dictionary (see Figure 1).

1. Dependency analysis of Japanese and English sentences.

2. Detection of Word/phrase correspondences.

3. Disambiguation of correspondences.

4. Handling of remaining words.

Among IWSLT05 20,000 training data, some pairs consists of two or more sentences. We utilized the pairs with the same number of Japanese sentences and English sentences, and separated them into one-to-one Japanese English sentence pairs. As a result, we utilized 21,412 sentence pairs.

We explain these alignment steps in detail.

2.1. Dependency Analysis of Japanese and English Sentences

Japanese sentences are converted into dependency structures using a morphological analyzer, JUMAN, and a dependency analyzer, KNP [3]. These tools can detect Japanese sentence structures in high accuracy: for the news article domain, 99% for segmentation and POS-tagging, and 90% for dependency analysis. They are robust enough to handle travel domain conversations and the accuracy is almost the same with news article sentences.

Japanese dependency structure consists of nodes which correspond with content words. Function words such as postpositions, affixes, and auxiliary verbs are included in content words' nodes.

For English sentences, Charniak's nlparser is used to convert them into phrase structures [4], and then they are transformed into dependency structures by rules defining head words for phrases. In the same way as Japanese, each content word composes a node of English dependency tree.

Charniak's nlparser was trained on Penn Treebank, and is not necessarily suitable for travel domain conversations. In some cases, basic English sentences were wrongly parsed by the parser.

2.2. Detection of Word/Phrase Correspondences

Japanese word/phrase to English word/phrase correspondences are detected by two methods.

One is to use a Japanese-English dictionary, EIJIRO [5]. The original EIJIRO contains about 1.5M entries, but we utilized about 0.9M entries excluding slang words/expressions.

The other method handles transliteration. For possible person names and geo names suggested by the morphological analyzer and Katakana words (Katakana is a Japanese alphabet usually used for loan words), their possible transliterations are produced and their similarity with words in the English sentence is calculated based on the edit distance. If there are similar words exceeding the threshold, they are handled as a correspondence.

For example, the following words can be corresponded by the transliteration module, which are rarely handled by the existing bilingual dictionary entries:

新宿 → Shinjuku ↔ Shinjuku (similarity:1.0)
ローズワイン → rosuwain ↔ rose wine (similarity:0.78)

The units of correspondences are nodes, and function words in nodes are included in the correspondences of content words. If the bilingual dictionary and transliteration module detect a correspondence with two or more content words, the correspondence of two or more nodes are generated accordingly. In Figure 1, for example, the two Japanese nodes "交差 (cross)" and "点 (point) で" corresponds to the one English node "at the intersection".

2.3. Disambiguation of Correspondences

The method described in the previous section sometimes detects ambiguous correspondences, that is, one-to-many or many-to-many correspondences. Such ambiguity is resolved based on harmonious criteria.

Suppose there is a correspondence X with ambiguity, and there is an unambiguous correspondence Y with the distance n in the Japanese dependency tree and the distance m in the English dependency tree, we give the score $1/n + 1/m$ to the correspondence X, since we can consider that the nearer Y is to X, the more strongly Y supports X. Here we define the distance of correspondences as the number of traversing nodes in a dependency tree. For example, in Figure 1, the distance between "the car" and "came" is 1, and that between "the car" and "at the intersection" is 2.

Then, we accept the ambiguous correspondence, the sum of whose neighboring correspondences' scores is the largest, and reject the others conflicting with the accepted one. This calculation is repeated until all the ambiguous correspondences are resolved.

IWSLT05 training sentences are fairly short, and most correspondences are unambiguous. Ambiguous correspondences are only 4.8%.

2.4. Handling of Remaining Words

The alignment procedure so far found all corresponds in parallel sentences. Then, we merge the remaining nodes into existing correspondences.

First, the root nodes of the dependency trees are handled as follows. In the given training data, we suppose all parallel sentences have appropriate translation relation. Accordingly, if neither root nodes (of the Japanese dependency tree and the English dependency tree) are included in any correspondences, the new correspondence of the two root nodes are generated. If either root node is remaining, it is merged into the correspondence of the other root node.

Then, both for Japanese remaining node and English remaining node, if it is within a base NP and another node in the NP is in a correspondence, it is merged into the correspondence. The other remaining nodes are merged into correspondences of their parent (or ancestor) nodes.

In the case of Figure 1, "あの (that)" is merged into the correspondence "車 (car) ↔ the car", since it is within an NP. Then, "突然 (suddenly)", "at me" and "from the side" are merged into their parent correspondence, "飛び出して来たのです (rush out) ↔ came".

We call the correspondences constructed so far as *basic correspondences*.

2.5. Comparison with EM based Alignment

Here, let us compare our alignment method with an EM based alignment. We tested an EM based tool, giza++ for the alignment of 20,000 training data [6]. We found many inappropriate word alignments in the giza++ results, and concluded that this size of training data might be too small for EM based alignment.

On the other hand, our method using a 0.9M-entry bilingual dictionary and a transliteration module could find correspondences quite accurately. For the given training set, we could conclude that our proposed method is superior to the EM based method.

However, the correspondence statistics in the whole training data must be an important information, and it is our future target to use a flat bilingual dictionary and the statistical information together.

2.6. Translation Example Database

Once we detect basic correspondences in the parallel sentences, all basic correspondences and all combination of adjoining basic correspondences (both in Japanese and English dependency trees) are registered into the translation example database.

From the parallel sentences in Figure 1, the three basic correspondences and their combinations such as "交差点で, 突然飛び出して来たのです ↔ came at me from the side at the intersection" and "突然あの車が飛び出して来たのです ↔ the car came at me from the side" are registered.

3. Translation

In the translation process, first, a Japanese input sentence is converted into the dependency structure as in the parallel sentence alignment. Then, translation examples for each subtrees are retrieved, the best translation examples are selected, and their English expressions are combined to generate the English translation (Figure 2).

3.1. Retrieval of Translation Examples

At first, the root of the input sentence is set to the retrieval root, and each sub-tree whose root is the retrieval root is retrieved step by step. If there is no translation example for a sub-tree, the retrieval for the current retrieval root stops. Then, each child node of the current retrieval root is set to the new retrieval root and its sub-trees are retrieved.

In the case of Figure 2, sub-trees from the root node "でした (was)" are retrieved: "でした (was)", "青 (blue) でした (was)", "信号 (signal) は でした (was)", "信号 (signal) は 青 (blue) でした (was)" and so on. Then, sub-trees from "青 (blue)" and sub-trees from "信号 (signal) は" are retrieved step by step.

If no translation example is found for a Japanese node, the bilingual dictionary is looked up and its translation is used as if it is an translation example. (If there is no entry in the dictionary we output nothing for the node.)

3.2. Selection of Translation Examples

Then, out of retrieved translation examples, good ones are selected to generate the English translation.

The basic idea of example-based machine translation is to prefer to use larger translation example, which takes into consideration larger context and could provide an appropriate translation. According to this idea, our system also selects larger examples.

The criterion is based on the size of translation example (the number of matching nodes with the input), plus the similarities of the neighboring outside nodes, ranging from 0.0 to 1.0 depending on the similarity calculated by a thesaurus. The similar outside node is used as a bond to combine two translation examples, as explained in the next section.

For example, if the size of a translation example is two, and the outside parent node is similar to the outside parent node of the matching Japanese input sub-tree by 0.3 similarity, and one outside child node is also similar to the corresponding input by 0.4, the score of the translation example becomes 2.7. [1]

The set of translation examples just enough for the input is searched in a greedy way. That is, the best translation example is selected among all the examples first, and then the next best example is selected for the remaining input nodes, and this process is repeated.

3.3. Combination of Translation Examples

It is easy to generate an English expression from a translation example, because it contains enough information of English dependency structure and word order. The problem is how to combine two or more translation examples.

[1] We proposed a method of selecting translation examples based on translation probability [7]. Though we used size and similarity based criteria for IWSLT05 because of time constraint, we are planning to use probability based criteria from now on.

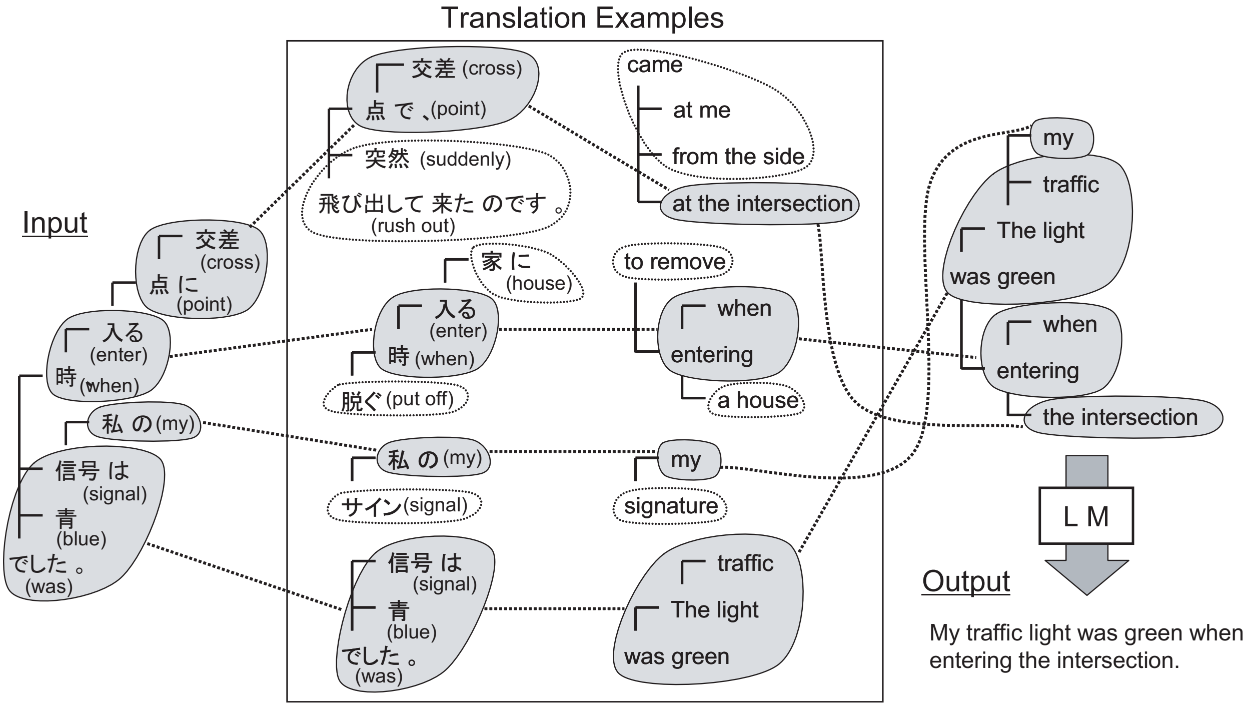

Figure 2: An example of Japanese-English translation.

However, in most cases, the bond node is available outside the example, to which the adjoining example is attached. There are two types of bond nodes: a child bond and a parent bond.

If there is a child node, it is easy to attach the adjoining example on it. For example, in Figure 2, the translation example “入る (enter) 時 (when)” has a child bond, “家 (house) に”, corresponding to “a house” in the English side. The adjoining example “交差点 (で) ↔ (at) the intersection” is attached on “家に”, which means “house” is replaced with “the intersection”.

On the other hand, a parent bond tells that the translation example modifies its head from the front or from behind, but there is no information about the order with the other children. Currently, we handle it as the first child if it modifies from the front; as the last child if it modifies from behind. In Figure 2, “私の ↔ my ” has a parent bond, “サイン ↔ sign” and it tells that “my” should modify its head from the front. Then, “my” is put to the first child of “the light”, before “traffic”.

It is not often, but if there is no bond, the order of combining two translation examples is controlled by heuristic rules.

3.4. Handling of Numerals

Numerals in Japanese are translated into English in several ways.

- cardinal : 124 → one hundred twenty four
- ordinal (e.g., day) : 2 日 → second
- two-figure (e.g., room number, year) : 124 → one twenty four
- one-figure (e.g., flight number, phone number) : 124 便 → one two four
- non-numeral (e.g., month) : 8 月 → August

At the time of parallel sentence alignment, it is checked in which type Japanese numerals are translated.

Translation examples of non-numeral type are used only if the numerals match exactly (“8 月 → August” cannot be used to translate “7 月”). However, translation examples of the other types can be used by generalizing numerals, and the input numeral is transformed according to the type. For example, “2 日 → second” can be used to translate “13 日”, transforming to the ordinal, “thirteenth”.

4. Handling of Pronoun Omission

In Japanese-English translation, omission of pronouns often causes problems. In conversational utterances, Japanese pronouns such as “私 (I)”, “あなた (you)”, “これ (this)” are often omitted, and this could cause erroneous translations. Essentially, omissions in Japanese sentences should be analyzed appropriately (in the case of parallel sentences, referring to English translations). However, the current system handles this problem using a language model of English.

Table 1: Evaluation results.

	BLEU	NIST
Development 1	0.4245	8.5655
Development 2	0.4056	8.4967
IWSLT05 manual	0.3718	7.8472
IWSLT05 ASR	0.3361	7.4157

There are two patterns when pronoun omission causes erroneous translations. One is that a pronoun is omitted in a translation example and not omitted in an input sentence. In such a case, there is no correspondence for the English pronoun, and it is merged into the other (usually predicate's) correspondence. If this merged pronoun is used in the translation, it overlaps with the pronoun from the input. For example, if the translation example "胃 (stomach) が 痛いのです (ache) ↔ I 've a stomachache" is used to translate "私 (I) は胃 (stomach) が 痛いです (ache)", the translation becomes "I I 've a stomachache" naively. To solve this problem, the merged pronoun is marked at the alignment, and two translations with it and without it are generated and ranked using a language model of English.

The opposite case also causes erroneous translations. That is, when a pronoun is in a translation example and is omitted in an input, the ungrammatical English sentence without pronoun is generated. For example, when "これ (this) を 日本 (Japan) へ 送って下さい (mail) ↔ will you mail this to Japan" is used to translate "日本 (Japan) へ 送って下さい (mail)", the translation becomes "will you mail to Japan" by eliminating "これを ↔ this". To handle such a problem, a bond node, which is not used for translation in a normal case, is used as a translation candidate when the bond node is a pronoun, and the best translation is selected using a language model of English.

In the IWSLT05, we used English sentences in 20,000 training data and Cam_Toolkit by CMU for a English language model [8].

5. Results

Our Japanese-English translation system challenged to both manual manuscript translation and ASR output translation (for ASR output we just translated the best path, though). Our system utilized Japanese and English parsers and a bilingual dictionary, and it was categorized to "supplied & tools" data track.

Table 1 shows evaluation scores for development set 1, development set 2, and the test set. Since we have not overtuned our system to development sets, IWSLT05 test set might be a bit tough task, which means that the coverage by training set is a bit small.

When our system translates one test sentence (7.5 words/3.2 nodes on average), 1.8 translation examples of the size of 1.5 nodes, and 0.5 translation from the bilingual dictionary are used.

6. Discussion

We examined the translation results and found out that it was not the case that there was a few major problems, but there were variety of problems, such as parsing errors of both languages, excess and deficiency of the bilingual dictionary, and the inaccurate and inflexible use of translation examples.

Now, let us discuss the biggest question: "is the current parsing technology useful and accurate enough for machine translation?"

If the translation performance was significantly better than the other systems without parsing, we could answer "YES" to the question. However, unfortunately our performance is average and we cannot claim that. Currently, we can at least dispel the suspicion that parsing might cause side-effects and lower translation performance.

As we mentioned above, parsing errors are not a principal cause of translation errors, but these are not a few. One of the possible countermeasures is to reconsider the learning process of an English parser. The English parser used here is learned from Penn Treebank, and seems to be vulnerable to conversational sentences in travel domain.

Furthermore, it is quite possible to improve parsing accuracies of both languages complementarily by taking advantage of the difference of syntactic ambiguities between the two languages [9]. This approach may not substantially improve the parsing accuracy of the travel domain sentences, because of their short length, but is promising for translating longer general sentences.

7. Conclusion

As we stated in Introduction, we not only aim at the development of machine translation through some evaluation measure, but also tackle this task from the comprehensive viewpoint including the development of structural NLP. The examination of translation errors revealed the problems, such as problems in parsing and inflexible matching of a Japanese input and Japanese translation examples. Resolving such problems is considered to be an important issue not only for MT but also for other NLP applications. We pursue the study of machine translation from this standpoint continuously.

8. References

[1] M. Nagao, "A framework of a mechanical translation between Japanese and English by analogy principle," in *Proceedings of the international NATO symposium on Artificial and human intelligence*, 1984, pp. 173–180.

[2] F. J. Och, C. Tillmann, and H. Ney, "Improved alignment models for statistical machine translation," in *Proceedings of the Joint Conference of Empirical Methods in Natural Language Processing and Very Large Corpora*, 1999, pp. 20–28.

[3] S. Kurohashi and M. Nagao, “A syntactic analysis method of long Japanese sentences based on the detection of conjunctive structures,” *Computational Linguistics*, vol. 20, no. 4, pp. 507–534, 1994.

[4] E. Charniak, “A maximum-entropy-inspired parser,” in *Proceedings of the 1st Meeting of the North American Chapter of the Association for Computational Linguistics*, 2000, pp. 132–139.

[5] Electronic Diciotionary Project, *EIJIRO 2nd Edition*. ALC Press Inc., 2005.

[6] F. J. Och and H. Ney, “A systematic comparison of various statistical alignment models,” *Computational Linguistics*, vol. 29, no. 1, pp. 19–51, 2003.

[7] E. Aramaki, S. Kurohashi, H. Kashioka, and H. Tanaka, “Probabilistic model for example-based machine translation,” in *Proceedings of MT Summit X*, 2005, pp. 219–226.

[8] P. Clarkson and R. Rosenfeld, “Statistical language modeling using the CMU-Cambridge toolkit,” in *Proceedings of the European Conference on Speech Communication and Technology*, 1997, pp. 2707–2710.

[9] Y. Matsumoto, H. Ishimoto, and T. Utsuro, “Structural matching of parallel texts,” in *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, 1993, pp. 23–30.