A Hybrid Rule and Example-based Method for Machine Translation*

Satoshi Shirai, Francis Bond and Yamato Takahashi
NTT Communication Science Laboratories
1-1 Hikari-no-oka, Yokosuka-shi, Kanagawa-ken, Japan 239
{shirai,bond,yamato}@cslab.kecl.ntt.co.jp

Abstract

This paper introduces a new example-based method of machine translation in which the examples need not be direct translations. The system will weed out strange examples during translation, allowing the use of currently available sentence aligned corpora as data. Rule-based modules are used where appropriate. A prototype Japanese-to-English system has been implemented that allows multiple users to share corpora.

1 Introduction

Methods for machine translation can be generally classified as rule-based or example-based, and each has numerous problems which remain unsolved. Especially in Japanese-to-English translation, due to the difference in language categories, current methods are far from being at the stage where they can be of practical use (Narita 1996). This paper will attempt to make use of the strengths of both the rule and example-based methods to suggest a form of machine translation that can be used with existing technology.

We will first discuss the strengths and weaknesses of various translation methods.

1.1 Rule-Based Translation

Most of the machine translation software on the market today is rule-based. These systems consist of (1) a process of analyzing input sentences (morphological, syntactic and/or semantic analyses) and (2) a process of generating sentences as a result of a series of structural conversions based on an internal structure or some interlingua. The steps of each process are controlled by the dictionary and the rules.

As the accuracy of translation by the system is the product of the accuracies of each process, it is necessary to enlarge the magnitude and to upgrade the precision of existing dictionaries and rules for each step (Ikehara et al. 1993), and this is extremely labor intensive.

Further, in-depth analyses enable the use of long-distance relationships and related information yet they tend to lose the collocational relations between words. In addition, most text produced by rule-based methods is incohesive. This is for two reasons, (1) the rules needed to increase cohesion are not yet fully understood and (2) those that are understood often rely on a full semantic and pragmatic analysis of the text, which is rarely available.

1.2 Example-based Translation

To overcome the problems of dictionaries and rules in the rule-based translation method, a method of translation by the principle of analogy has been proposed (Nagao 1984). This is done by collecting aligned translated example sentences and translating the input sentence by imitating the translation of a sentence that resembles it. This has its merits in that, as long as there is a translated example, a well structured translation will be generated. There is no need to prepare dictionaries and rules through individual analysis of linguistic phenomena. Improvement in the translation capability could be expected by merely adding examples of translations. This has resulted in a large amount of research in this field.

This analogy method, however, frequently assumes the existence of an aligned corpus with examples that align on a strict 1-to-1 basis as well as on appropriate tag information showing the correspondence between words and phrases (Sadler 1989:117). Yet, cases of direct translation equivalents are limited in number and assessments of similarity using a thesaurus are rarely reliable. Example-based translation that relies on a corpus with very similar text is really just a variation on translation memory, very useful for tasks such as upgrading manuals, where much of

*This paper was read at the 4th Natural Language Processing Pacific Rim Symposium 1997: NLPRS-97; Phuket, Thailand and appears in the proceedings: pp 49–54.

the text is reused, but not generally useful. And even if it were possible to secure a large volume of corresponding example sentences, the task of tagging them in a uniform and accurate manner is difficult. For example, Kaji et al. (1992) make a set of templates with variable expressions from their corpus, before using it. These templates are only be as accurate as the parsers used to prepare them, and must be remade every time the parsers change.

Cranias et al. (1995) propose a matching method based on differences between function and content words. It relies crucially however, on segmenting sentences into coherent segments and alignment at the sub-sentential level, both processes that are hard to automate.

1.3 Combined Translation Methods

Multi-engine systems use both the rule and example-based methods and then choose one of the translations. These systems take over all of the strengths and weaknesses of each method and add the fresh problem of how to decide which output to use.

Brown (1996) is an example of an example-based system run in parallel with a rule-based (knowledge-based) system as part of the Pangloss system. It only translates sequences of connected words, and so fails to give one of the expected benefits of an example-based system, the production of sentences with coherent structure. Its main strength seems to be in the fact that it is easy to adapt to new languages.

2 A Hybrid Translation Method

The following hybrid design is an effort to produce a method that makes the most of the strengths of both methods and that will compensate for their weaknesses. The strengths of the rule-based method lie in the fact that information can be obtained through introspection and analysis, while those of the example-based method are that correspondences can be found from raw data. The weakness of the rule-based method is that the accuracy of the entire process is the product of the accuracies of each sub-stage. The weakness of the example-based method is the difficulty of finding appropriate examples.

The basic outline of the algorithm is as follows:

1. Select a set of candidate sentences which are similar to the input sentence (§2.2)
2. Select the most typical translation out of those corresponding to the candidates (§2.3)
3. Use this translation and its source as templates to translate the input sentence (§2.4)

The major innovation of this algorithm is in step two. Instead of simply choosing the source-target pair whose source sentence best matches the input sentence, a pair is chosen which both matches the input sentence and has a translation similar to other examples. By discarding candidates with atypical translations, the algorithm filters out free, incorrect or context dependent translations. This means that the input corpus does not have to consist of good, context-independent sentence pairs to be useful. The only requirement is that there be enough translations to be able to find a typical translation.

A variety of methods can be used to determine similarity in step one, to select the most typical translation in step two, and to finally translate the sentence in step three. The methods currently being used in our system are described in the following subsections, after a brief discussion of the construction of the corpus.

As all processing is done during the translation process, improvements in any of the modules will not require the entire corpus to be re-parsed.

A more detailed outline is given in Figure 1.

2.1 Creating and Indexing a Corpus

A practical example-based system requires a large volume of suitable data. In the case of our hybrid algorithm, for the corpus to be suitable it need only consist of sentences that are loosely aligned, not necessarily exact translations.

A large volume of newspaper data is currently available, for example, the Nihon Keizai Shinbun,[1] much of it on CD-ROM. There is much more Japanese data than English. Considering stock-market reports alone, we estimate that there are 1,000,000 Japanese sentences (35 characters on average) and 150,000 English sentences (13 words on average) each year. The Japanese and English articles are not direct translations of each other, but for around half of the English sentences there are Japanese sentences that are close to being literal translations (Shirai et al. 1995). The aligned sentence pairs are, to some extent, translation equivalents. Ideally they are sentences with equivalent meanings in the two languages, in reality many of them only contain sub-sections with equivalent meanings. Note that it is not important which was originally the source and which the translation.

As our algorithm weeds out unsuitable sentences during translation, we focus on recall rather

[1] A Japanese financial newspaper.

For each input sentence: S_I

1. Find candidate sentences $\{S_i$: S_i is similar to $S_I\}$ (§2.2)

   If there are none, translate using rule-based system to give T_I, goto step 4.

2. Select the template: S_t (§2.3)

   (a) Rank the candidates, S_i, by similarity to the input sentence

   (b) Cluster the translations, T_i, of the candidate sentences

   (c) Select the highest ranked pair of the best cluster (S_t, T_t)

3. Translate S_I by analogy to S_t (§2.4)

   For each difference d_i between input S_I and selection S_t

   (a) Find the corresponding section t_i of the selected translation T_t

   (b) Replace t_i with the translation of d_i, translated using the rule-based modules

   Adjust for number agreement and so forth

4. Output the adjusted sentence T_I

Figure 1: An outline of the hybrid algorithm

than precision when aligning our sentences, which can thus be done entirely automatically. First we align the newspaper articles, using numerical expressions and proper nouns as anchors following the method outlined in Takahashi et al. (1997). Then we align sentences within the aligned articles. We accept as aligned sentences, ones that contain even a small amount of equivalent text, so sentence level alignment can also be done automatically. Currently we adopt a method that uses both statistical and dictionary information (Haruno & Yamazaki 1996), but any method could be used.

We have tagged the data with SGML tags, using the TEI P3 document type definition (Sperberg-McQueen & Burnard 1994). We do not tag any elements smaller than sentences as our algorithm does not require a corpus tagged with details of the internal structure.

Each article and each sentence in the article has a unique ID. The Japanese and English data are stored in separate files. A separate file contains links showing the correspondences between the two languages. The format is similar to that used in the **Lingua** Project (Bonhomme & Romary 1995). As the links are in a separate file, they can easily be replaced as better alignment algorithms appear and are used.

In addition, an index of all n-grams (n $\geq$2) that appear more than once in the Japanese data has been prepared. It is used for finding similar sentences. The n-grams are found using the method outlined in (Ikehara et al. 1996), which eliminates any n-grams that appear only as substrings of larger n-grams. For a language such as English, which is separated into words by default, a word index (preferably lemmatized), could also be used. We used a variety of trie index, for fast and efficient searching (Aoe et al. 1992).

We end up with the following data:

- Source Sentences S^i
- Target Sentences T^j
- links l_{i-j} (not all sentences have links)
- trie index (of all n-grams that appeared more than once in the source sentences)

2.2 *Finding Candidate Sentence Pairs*

The input sentence S_I is searched for any n-grams that appear in the index (that is n-grams that appeared more than once in the Japanese corpus). All sentences in the corpus that contain one or more of the n-grams, and have English equivalents, are selected as candidate sentences.

For example, consider the following input sentence, which contains the following indexed n-grams:[2] **nikkei, heikin, gatsu-mono-wa-zokuraku, wa-zokuraku**.

S_I nikkei heikin 10 gatsu mono wa
Nikkei average 10 month thing TOP
zokuraku .
continue-decline .

The Nikkei Average October contracts continued declining

For the sake of our explaining the method, we will assume it only matched the following three sentences (matching n-grams marked bold):

(1) **nikkei heikin** 9 **gatsu mono wa**
Nikkei average 9 month thing TOP
zokuraku .
continue-decline .

[2]To make an example with few enough matches to show, we have created examples assuming an unreasonably small corpus, in reality there would be more indexed n-grams. All input sentences are actually stored using Japanese characters, although we show only their transliterations in this paper.

The Nikkei Average September contracts were lower.

(2) **nikkei** tentoo **heikin wa**
Nikkei over-the-counter average TOP
zokuraku .
continue-decline .

The Nikkei over-the-counter average continued declining

(3) 8 **gatsu mono wa zokuraku** .
8 month thing TOP *continue-decline* .

August contracts continued declining.

However, if there are no candidate sentences, the hybrid design method is unable to function and the sentence is translated using a rule-based translation system.

2.3 Selecting the template

2.3.1 Ranking the Candidate Sentences

The input sentences and all the candidates are segmented using a morphological analyzer (the same as used by the rule-based Japanese-to-English machine translation system **ALT-J/E** (Ikehara et al. 1991)).

The sentences are ranked using the similarity metric given below.

The result is an ordered set of candidate sentences and their translation equivalents $S_1, \ldots S_n$, the higher ranked a sentence is, the more similar it is to the input sentence.

At this stage it would be possible to discard some lower ranked sentences, in order to increase speed, either by discarding any with similarities below a certain threshold,[3] or all those beyond a certain number. However, at present, speed has not been a problem, so we do not do this.

Definition of 'similarity'

The similarity metric is calculated with two components, one based on the order of shared segments (M_o), and one based only on their co-occurrence (M_c).

The two components are calculated as follows. First all segments of S_I are given a value from left to right, starting with 0, and increasing by one for each segment. Then all segments in S_i that also occur in S_I are given the same values, but differing segments are given a very high value (such as 99).

M_o and M_c are defined as follows (bubble sort values given when compared to S_I):

$$M_o(S_I, S_i) = \frac{\textit{no. of swaps to bubble sort } S_i}{\textit{no. of swaps to reverse } S_i} \quad (1)$$

[3] If there were no candidates above this threshold, the sentence would be translated by the rule-based system

$$M_c(S_I, S_i) = \frac{\textit{number of shared segments}}{\textit{number of segments in } S_i} \quad (2)$$

The combined similarity metric $M = (1 - M_o)M_c$. M_o penalizes changes in order, while M_c penalizes non-matching segments. We take the final similarity as the average of $M(S_I, S_i)$ and $M(S_i, S_I)$. No explicit semantic analysis is undertaken.

For example the above sentences are ranked as follows:

S_I nikkei heikin 10 gatsu mono wa zokuraku .
0 1 2 3 4 5 6 7

S_1 nikkei heikin 9 gatsu mono wa zokuraku .
0 1 99 3 4 5 6 7
($M(S_I, S_i)$ 5 swaps, 7 shared segments)

S_2 8 gatsu mono wa zokuraku .
99 3 4 5 6 7
($M(S_I, S_i)$ 5 swaps, 5 shared segments)

S_3 nikkei-tentoo-heikin wa zokuraku .
99 5 6 7
($M(S_I, S_i)$ 3 swaps, 3 shared segments)

Note that *nikkei-tentoo-heikin* "the Nikkei over-the-counter average" appears as a single word in the user dictionary, and thus is analyzed as a single constituent. Therefore it does not match *nikkei* "Nikkei" or *heikin* "average", although it matched them using the initial n-gram index.

This method considers not only the number of matching characters and the number of continuous matching characters, as in Sato (1992), but also the number of non-matching characters found in the candidates.

Because our morphological analyzer also gives POS and sense, as well as chunking the strings into phrases we are currently investigating ways of using this additional information, for example by comparing the number and order of case-marked noun phrases and comparing senses. However, our method is designed to work with a large corpus, so a simple fast algorithm is to be preferred.

2.3.2 Finding the best cluster

The next step is to cluster the translation equivalents $\{T_i\}$ of the candidate sentences $\{S_i\}$ into groups of similar sentences. This is done in order to find translations that are reasonably direct. As all the candidate sentences are similar to the input sentence, we expect that direct translations will also be similar.

We cluster the translations as follows:

1. Determine the frequency of all words in the set of translation equivalents $\{T_i\}$, ignoring

any words that are in a set of stop words (mainly function words such as articles, auxiliary verbs and prepositions).

2. If there is one most common word, then the best cluster is the set of sentences that include it. Otherwise, the best cluster is the set of those with the maximum number of the most common words.

For example (frequencies given as subscripts; stop words have no subscripts):

T_1 The Nikkei$_2$ Average$_1$ September$_1$ contracts$_2$ were lower$_1$

T_2 August$_1$ contracts$_2$ continued$_2$ declining$_2$

T_3 The Nikkei$_2$ over-the-counter$_1$ average$_1$ continued$_2$ declining$_2$

In this case the best cluster has two members T_2 and T_3.

2.3.3 Selecting the sentence with the most typical translation

The sentence pair to be used in the actual translation is the sentence pair (candidate sentence + translation equivalent) in the best cluster that has the highest similarity to the input sentence (in our example (S_2, T_2)). We call the two sentences the source template (S_t) and the target template (T_t). This sentence pair ideally[4] has the following properties: (1) the source template resembles the input sentence, (2) the translation template is a reasonably direct translation of the source template. This makes the sentence pair a suitable template for example-based translation by analogy.

2.4 Translating the input, using the template as guide

The differing sections of the input sentence and the source template are identified. Translations of these different sections in the input sentence are produced by rule-based methods and these translated sections are fitted into the translation sentence in the template. The resulting sentence is then smoothed over, by checking for agreement and inflection mismatches.

The resulting translation has the overall structure provided by the example-based template, with only individual words or phrases, translated by the rule-based system. In general, rule-based systems give better results for small segments, so this gives the best of both worlds.

The input sentence and selected template are:

S_I nikkei heikin 10 gatsu mono wa zokuraku .

S_t 8 gatsu mono wa zokuraku .

T_t August contracts continued declining.

2.4.1 Finding the differences

In order to find any differences, both the input sentence and source template are parsed, and their parses are compared. Because the sentences are similar, even if there are errors in the parse, they are often the same errors for both sentences and have little effect on finding the differences, so the algorithm will work with imperfect parsers. This is the only part of the process that requires parsing, and it is tolerant of errors, so there is no need for structural tagging in the corpus.

In this case the difference is between *nikkei heikin 10* and *8*. The parsing process however shows that *nikkei heikin 10 gatsu mono* is all one noun phrase, and so matches it with *8 gatsu mono* "August contracts"

2.4.2 Replacing the differences

The rule-based translation of the source template is compared with the target template. Those parts of the template that correspond to the differing sections are replaced by their translations.

nikkei heikin 10 gatsu mono is translated as "The Nikkei Average October contracts", by the rule-based noun phrase translation system, which gives good results for small segments. This is slotted into T_t giving the following:

T_I The Nikkei Average October contracts continued declining.

2.4.3 Smoothing the output

A very rough surface analysis of the output sentence is used to check for person and number agreement.

At present we have no mechanism for identifying and deleting extraneous elements in the example-based translation, such as temporal adverbs. This is one of the major sources of errors in our system. We are currently adding a filter to identify and delete such terms.

2.5 System Architecture

The features of the system as it is implemented are listed below:

- The system operates as a multiuser client/server network. The server runs on UNIX and the clients on Windows-NT.
- Users can combine their own aligned corpora with the system's, and share them amongst themselves.

[4] If the corpus was large enough to produce a choice of candidates.

- The rule-based components and dictionaries are taken from **ALT-J/E**.

The prototype was tested translating from Japanese to English, with a corpus of 5,000 sentences[5]. We are now in the process of testing it with a year's data.

3 Conclusion

The merits of using examples are that, translations of expressions which are idiomatic, literal or domain dependent can all be put to use; and the system should be reversible, that is it can translate in either direction. Previously, many example-based systems assumed the existence of large word or phrase aligned corpora. To date, we know of no large scale corpora accurately aligned below the sentence level. For an example-based system to be useful in the foreseeable future, it has to be able to accept loosely aligned corpora, not aligned at low levels. We have suggested a design for a system that can use such loosely aligned texts. We have implemented a prototype of such a system, that works using corpora of the level currently available, to translate from Japanese to English. The prototype allows users to take advantage of any aligned text they may already have by adding it to the set of sentences searched by the system.

In the future, we plan to improve the individual modules in the system, in particular the similarity measure and output smoother.

Acknowledgments

We would like to thank Satoru Ikehara, Toshiaki Tanabe, Masatoshi Tachibana, Hiroko Inoue, Makiko Nishigaki, Hiromi Ueda and Tim Baldwin for their help and advice.

References

Aoe, J., K. Morimoto & T. Sato: 1992, 'An efficient implementation of trie structures', *Software Practice & Experiments*, **22**(9): 695–721.

Bonhomme, Patrice & Laurent Romary: 1995, 'The lingua parallel concordancing project: Managing multilingual texts for educational purpose', in *Language Engineering '95*.

Brown, Ralf D.: 1996, 'Example-based machine translation in the pangloss system', in *16th International Conference on Computational Linguistics: COLING-96*, pp. 125–130.

Cranias, Lambros, Harris Papageorgiou & Stelios Piperidis: 1995, 'A matching technique in example-based machine translation', http://xxx.lanl.gov/abs/cmp-lg/9508005).

Haruno, Masahiko & Takefumi Yamazaki: 1996, 'High-performance bilingual text alignment using statistical and dictionary information', in *34th Annual Conference of the Association for Computational Linguistics*, pp. 131–138.

Ikehara, Satoru, Masahiro Miyazaki & Akio Yokoo: 1993, 'Classification of language knowledge for meaning analysis in machine translation', *Transactions of the Information Processing Society of Japan*, **34**(8): 1692–1704, (in Japanese).

Ikehara, Satoru, Satoshi Shirai & Hajime Uchino: 1996, 'A statistical method for extracting uninterrrupted and interrupted collocations from very large corpora', in *16th International Conference on Computational Linguistics: COLING-96*, Copenhagen, pp. 574–579.

Ikehara, Satoru, Satoshi Shirai, Akio Yokoo & Hiromi Nakaiwa: 1991, 'Toward an MT system without pre-editing – effects of new methods in **ALT-J/E**–', in *Third Machine Translation Summit: MT Summit III*, Washington DC, pp. 101–106, (http://xxx.lanl.gov/abs/cmp-lg/9510008).

Kaji, H., Y. Kida & Y. Morimoto: 1992, 'Learning translation templates from bilingual text', in *14th International Conference on Computational Linguistics: COLING-92*, pp. 672–678.

Nagao, Makoto: 1984, 'A framework of mechanical translation between Japanese and English by analogy principle', in Elithorn & Banerji, eds., *Artificial and Human Intelligence*, Elsevier, pp. 179–180.

Narita: 1996, 'Language type and machine translation', *IPSJ SIG Notes 96-NL-114-21*, **96**(65): 143–50, (in Japanese).

Sadler, Victor: 1989, *Working with Analogical Semantics: Disambiguation techniques in DLT*, FORIS Publications.

Sato, Satoshi: 1992, 'CTM: An example based translation aid system', in *14th International Conference on Computational Linguistics: COLING-92*, pp. 1259–1263.

Shirai, Satoshi, Mitsue Matsuo, Takako Seshimo, Susumu Fujinami & Satoru Ikehara: 1995, 'Constructing an aligned Japanese/English corpus of newspaper articles (3) — peculiarities of the articles and sentence alignment —', in *Record of the 1995 Joint Conference of Electrical and Electronics Engineers in Kyushu*, p. 857, (in Japanese).

Sperberg-McQueen, C. M. & Lou Burnard, eds.: 1994, *Guidelines for Electronic Text Encoding and Interchange*, Oxford: Chicago.

Takahashi, Yamato, Satoshi Shirai & Francis Bond: 1997, 'A method of automatically aligning Japanese and English newspaper articles', in *Natural Language Processing Pacific Rim Symposium '97: NLPRS-97*, Phuket, pp. 657–660.

[5] The data was being collected as the system was being built.