EXAMPLE-BASED CORRECTION OF WORD SEGMENTATION AND PART OF SPEECH LABELLING

Tomoyoshi Matsukawa , Scott Miller, and Ralph Weischedel

BBN Systems and Technologies
70 Fawcett St.
Cambridge, MA 02138

ABSTRACT

This paper describes an example-based correction component for Japanese word segmentation and part of speech labelling (AMED), and a way of combining it with a pre-existing rule-based Japanese morphological analyzer and a probabilistic part of speech tagger.

Statistical algorithms rely on frequency of phenomena or events in corpora; however, low frequency events are often inadequately represented. Here we report on an example-based technique used in finding word segments and their part of speech in Japanese text. Rather than using hand-crafted rules, the algorithm employs example data, drawing generalizations during training.

1. INTRODUCTION

Probabilistic part of speech taggers have proven to be successful in English part of speech labelling [Church 1988; DeRose, 1988; de Marcken, 1990; Meteer, et. al. 1991, etc.]. Such stochastic models perform very well given adequate amounts of training data representative of operational data. Instead of merely stating what is possible, as a non-stochastic rule-based model does, probabilistic models predict the likelihood of an event. In determining the part of speech of a highly ambiguous word in context or in determining the part of speech of an unknown word, they have proven quite effective for English.

By contrast, rule-based morphological analyzers employing a hand-crafted lexicon and a hand-crafted connectivity matrix are the traditional approach to Japanese word segmentation and part of speech labelling [Aizawa and Ebara 1973]. Such algorithms have already achieved 90-95% accuracy in word segmentation and 90-95% accuracy in part-of-speech labelling (given correct word segmentation). The potential advantage of a rule-based approach is the ability of a human coding rules that cover events that are rare, and therefore may be inadequately represented in most training sets. Furthermore, it is commonly assumed that large training sets are not required.

A third approach combines a rule-based part of speech tagger with a set of correction templates automatically derived from a training corpus [Brill 1992].

We faced the challenge of processing Japanese text, where neither spaces nor any other delimiters mark the beginning and end of words. We had at our disposal the following:

- A rule-based Japanese morphological processor (JUMAN) from Kyoto University.
- A context-free grammar of Japanese based on part of speech labels distinct from those produced by JUMAN.
- A probabilistic part-of-speech tagger (POST) [Meteer, et al., 1991] which assumed a single sequence of words as input.
- Limited human resources for creating training data.

This presented us with four issues:

1) how to reduce the cost of modifying the rule-based morphological analyzer to produce the parts of speech needed by the grammar,
2) how to apply probabilistic modeling to Japanese, e.g., to improve accuracy to ~97%, which is typical of results in English,
3) how to deal with unknown words, where JUMAN typically makes no prediction regarding part of speech, and
4) how to estimate probabilities for low frequency phenomena.

Here we report on an example-based technique for correcting systemmatic errors in word segmentation and part of speech labelling in Japanese text. Rather than using handcrafted rules, the algorithm employs example data, drawing generalizations during training. In motivation, it is similar to one of the goals of Brill (1992).

2. ARCHITECTURE

The architecture in Figure 1 was chosen to minimize labor and to maximize use of existing software. It employs JUMAN first to provide initial word segmentation of the text, an annotation-based algorithm second to correct both segmentation errors and part of speech errors in JUMAN output, and POST third both to select among ambiguous alternative segmentations/part-of-speech assignments and also to predict the part of speech of unknown words.

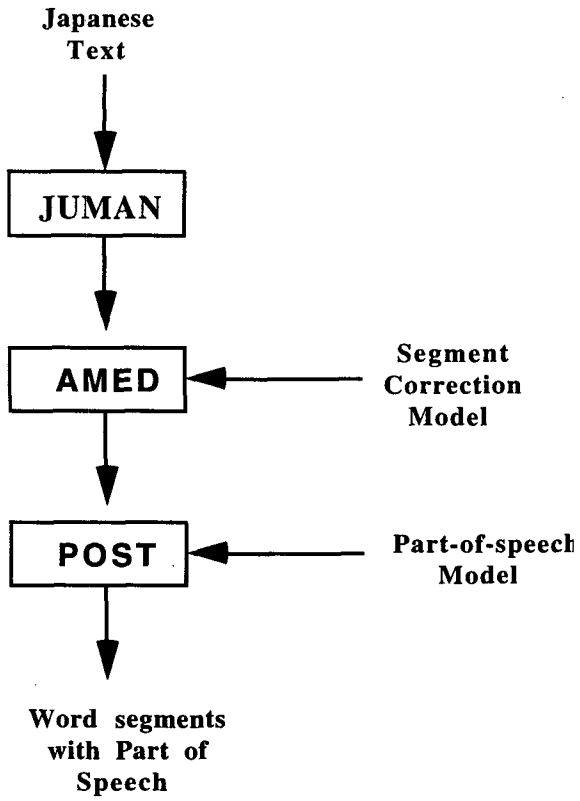

Figure 1: *Architecture*

Let us briefly review each component. JUMAN, available from Kyoto University makes segmentation decisions and part of speech assignments to Japanese text. To do this, it employs a lexicon of roughly 40,000 words, including their parts of speech. Where alternative segmentations are possible, the connectivity matrix eliminates some possibilities, since it states what parts of speech may follow a given part of speech. Where the connectivity matrix does not dictate a single segmentation and part of speech, generally longer words are preferred over shorter segmentations.

An example JUMAN output is provided in Figure 2. The Japanese segment is given first, followed by a slash and the part of speech. JUMAN employs approximately 45 parts of speech. [1]

海外旅行も楽しめる新型の旅行サービスを二月中旬から売り出す。

FIGURE 2a: *A Short Example Sentence*

海外/CN 旅行/SN も/TTM 楽/SN し/VB め/??? る/??? 旅行/SN サービス/SN を/CM 二月中旬/??? から/PT 売り出す/VB 。/KT

FIGURE 2b: *JUMAN output for example 2a above*

The correction algorithm (AMED) is trained with two parallel annotations of the same text. One of the annotations is JUMAN's output. The second is manually annotated corresponding to correct segmentation and correct part-of-speech assignments for each word. During training, AMED aligns the parallel annotations, identifies deviations as "corrections", and automatically generalizes these into correction rules. An example of automatic alignment appears in Figure 3.

AMED performs the following functions:

- Corrects some segmentation errors made by JUMAN.
- Corrects some part-of-speech assignment errors made by JUMAN. Some of these "corrections" actually introduce ambiguity which POST later resolves.
- Transforms the tag set produced by JUMAN into the tag set required by the grammar.

Note that all of these functions are the result of the learning algorithm, no rules for correction nor for translating JUMAN parts of speech into those for the grammar were written by hand.

The third component is POST, which assigns parts of speech stochastically via a Hidden Markov model, has been described elsewhere [Meteer, et al., 1991]. POST performs two vital functions in the case of our Japanese processing:

[1] CN = common noun; SN = sa-inflection noun (nominalized verb); VB = verb; VSUF = verb suffix; CM = case marker; etc.

- POST decides among ambiguous part-of-speech labellings and segmentations, particularly in those cases where AMED's training data includes cases where JUMAN is prone to error.

- POST predicts the most likely part of speech for an unknown word segment in context.

3. HOW THE ARCHITECTURE ADDRESSES THE ISSUES

In principle, a Hidden Markov Model implementation, such as POST, can make both part-of-speech decisions and segment text quite reliably. Therefore, why not just use POST; why use three components instead?

The clear reason was to save human effort. We did not have access to segmented and labelled Japanese text. Labelling tens of thousands (or even hundreds of thousands of words of text) for supervised training would have taken more effort and more time in a project with tight schedules and limited resources. JUMAN existed and functioned above 90% accuracy in segmentation.

A secondary reason was the opportunity to investigate an algorithm that learned correction rules from examples. A third reason was that we did not have an extensive lexicon using the parts of speech required by the grammar.

The architecture addressed the four issues raised in the introduction as follows:

1) AMED learned rules to transform JUMAN's parts of speech to those required by the grammar.

2) Accuracy was improved both by AMED's correction rules and by POST's Hidden Markov Model.

3) POST hypothesizes the most likely part of speech in context for unknown words, words not in the JUMAN lexicon.

4) The sample inspection method in AMED estimates probabilities for low frequency phenomena.

4. THE CORRECTION MODEL

The only training data for our algorithm is manually annotated word segmentation and part of speech labels. Examples of corrections of JUMAN's output are extracted by a procedure that automatically aligns the annotated data with JUMAN's output and collects pairs of differences between sequences of pairs of word segment and part of speech. Each pair of differing strings represents a correction rule; the procedure also generalizes the examples to create more broadly applicable correction rules.

JUMAN OUTPUT	DESIRED OUTPUT
海外/SN	海外/CN
旅行/SN	旅行/CN
も/TTM	も/TTM
楽/SN	楽し/VB
し/VB	
め/???	める/VSUF
る/???	
旅行/SN	旅行/CN
サービス/SN	サービス/CN
を/CM	を/CM
二月中旬/???	二月/CN
	中旬/CN
から/PT	から/PT
売り出す/VB	売り出す/VB
。/KT	。/KT

Figure 3a: *Alignment of JUMAN output with manually annotated correction data.*

楽/SN	楽し/VB
し/VB	
め/???	める/VSUF
る/???	
二月中旬/???	二月/CN
	中旬/CN

Figure 3b: *Pairs of differences collected from alignment in Figure 3a. above.*

We estimate probabilities for the correction rules via the sample inspection method. (see the Appendix.) Here, significance level is a parameter, from a low of 0.1 for ambitious correction through a high of 0.9 for conservative correction. The setting gives us some trade-off between accuracy and the degree of ambiguity in the results. One selects an appropriate value by empirically testing performance over a range of parameter settings. Correction rules are ordered and applied based on probability estimates.

When a rule matches, 1) AMED corrects JUMAN's output if the probability estimate exceeds a user-specified threshold, 2) AMED introduces an alternative if the probability falls below that threshold but exceeds a second user-supplied threshold, or 3) AMED makes no change if the probability estimate falls below both thresholds.

As a result, a chart representing word segmentation and part of speech possibilities is passed to POST, which was easily modified to handle a chart as input, since the underlying Viterbi algorithm applies equally well to a chart. POST then selects the most likely combination of word segmentation and part of speech labels according to a bi-gram probability model.

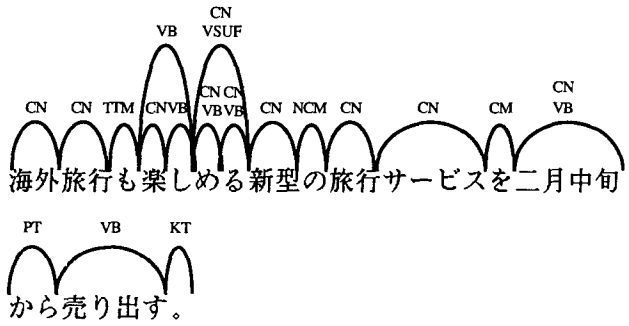

Figure 4: *Chart of alternatives produced by AMED.*

海外/CN 旅行/CN も/TTM 楽し/VB める/VSUF 新型/CN
の/NCM 旅行/CN サービス/CN を/CM 二月中旬/CN
から/PT 売り出す/VB 。/KT

Figure 5: *Final segmentation and labelling after POST.*

5. EXPERIENCE

The motivation for this study was the need to port our PLUM data extraction system [Weischedel, et al., 1992] to process Japanese text. The architecture was successful enough that it is part of (the Japanese version of) PLUM now, and has been used in Government-sponsored evaluations of data extraction systems in two domains: extracting data pertinent to joint ventures and extracting data pertinent to advances in microelectronics fabrication technology. It has therefore been run over corpora of over 300,000 words.

There are two ways we can illustrate the effect of this architecture: a small quanitative experiment and examples of generalizations made by AMED.

5.1 A Small Experiment

We ran a small experiment to measure the effect of the architecture (JUMAN + AMED + POST), contrasted with JUMAN alone. Japanese linguistics students corrected JUMAN's output; the annotation rate of an experienced annotator is roughly 750 words per hour, using the TREEBANK annotation tools (which we had ported to Japanese). In the first experiment, we used 14,000 words of training data and 1,400 words of test data. In a second experiment, we used 81,993 words of training data and a test set of 4,819 words.

Remarkably the results for the two cases were almost identical in error rate. In the smaller test (of 1,400 words), the error rate on part-of-speech labelling (given correct segmentation) was 3.6%, compared to 8.5%; word segmentation error was reduced from 9.4% to 8.3% using the algorithm. In the larger test (of 4,819 words), the error rate on part-of-speech labelling (given correct segmentation) was 3.4%, compared to 8.2%; word segmentation error was reduced from 9.4% to 8.3% using the algorithm.

Therefore, using the AMED correction algorithm plus POST's hidden Markov model reduced the error rate in part of speech by more than a factor of two. Reduction in word segmentation was more modest, a 12% improvement.

Error rate in part-of-speech labelling was therefore reduced to roughly the error rate in English, one of our original goals.

Both segmentation error and part of speech error could be reduced further by increasing the size of JUMAN's lexicon and/or by incorporating additional generalization patterns in AMED's learning alogrithm. However, in terms of improving PLUM's overall performance in extracting data from Japanese text, reducing word segmentation error or part-of-speech error are not the highest priority.

5.2 Examples of Rules Learned

One restriction we imposed on generalizations considered by the algorithm is that rules must be based on the first or last morpheme of the pattern. This is based on the observation in skimming the result of alignment that the first or last morpheme is quite informative. Rules which depend critically on a central element in the difference between aligned JUMAN output and supervised training were not considered. A second limitation that we imposed on the algorithm was that the right hand side of any correction rule could only contain one element, instead of the general case. Three kinds of correction rules can be inferred.

- A specific sequence of parts of speech in JUMAN's output can be replaced by a single morpheme with one part of speech.
- A specific sequence of parts of speech plus a specific word at the left edge can be replaced by a single morpheme with one part of speech.

- A specific sequence of parts of speech plus a specific word at the right edge can be replaced by a single morpheme with one part of speech.

The critical statistic in selecting among the interpretations is the fraction of times a candidate rule correctly applies in the training data versus the number of times it applies in the training. In spite of these self-imposed limitations in this initial implementation, the rules that are learned improved both segmentation and labelling by part of speech, as detailed in Section 5.1. Here we illustrate some useful generalizations made by the algorithm and used in our Japanese version of the PLUM data extraction system.

In example (1) below, the hyptohesized rule essentially recognizes proper names arising from an unknown, a punctuation mark, and a proper noun; the rule hypothesizes that the three together are a proper noun. This pattern only arises in the case of person names (an initial, a period, and a last name) in the training corpus.

1. */??? */KG */PN ===> PN

E/???	E・マークラッド/PN
・/KG	
マークラッド/PN	

Example (2) is a case where an ambiguous word ("nerai", meaning a"aim" or "purpose") is rarely used as a verb, but JUMAN's symbolic rules are predicting it as a verb. The rule corrects the rare tag to the more frequent one, common noun.

2. 狙い/VB ===> CN

狙い/VB	狙い/CN

Example (3) represents the equivalent of learning a lexical entry from annotation; if JUMAN had had it in its lexicon, no correction of segmentation (and part of speech) would have been necessary. There are many similar, multi-character, idiomatic particles in Japanese. Parallel cases arise in English, such as "in spite of" and "in regard to".

3. と/NCM */PT */CN */PT===> PT

と/NCM	との間で/PT
の/PT	
間/CN	
で/PT	

Example (4) is interesting since the rule learned corresponds to a general morphological phenomenon in Japanese. "Shita" converts an adverb to an adjective.

4. */ADV した/VB ===> ADJ

こう/ADV	こうした/ADJ
した/VB	

Example (5) represents a lexical omission where an inflected form, corresponding to the modal "can", is learned.

5. */??? る/??? ===> VSUF

め/???	める/VSUF
る/???	

6. CONCLUSION

The most interesting aspect of this work is the implementation and testing of a simple algorithm to learn correction rules from examples. Except for the annotation of text as to the correct data, the process is fully automatic. Even with as little data as we had initially (under 15,000 words), the learned correction rules improved the performance of morphological processing compared to the baseline system. Furthermore, though the original error rate of JUMAN was more than double the rate typically reported for stochastic part-of-speech labellers in English, the result of the correction algorithm plus our hidden Markov model (POST) reduced the error rate to a level comparable with that experienced in English. On the other hand, increasing the training data by a factor of five did not reduce the error rate substantially.

The architecture proposed is the morhpological component of the Japanese version of the PLUM data extraction system, and has been tested on more than 300,000 words of text in both a financial domain and a technical domain.

Hidden Markov Models, as implementd in POST, were applied to Japanese with relative ease. When additional data becomes available, we would like to test the performance of POST for both word segmentation and labelling part of speech in Japanese.

ACKNOWLEDGEMENTS

We wish to thank Professors Matsumoto and Nagao of Kyoto University who graciously made the JUMAN system available to us.

REFERENCES

1. Aizawa, T. and Ebara, T. (1973) "Mechanical Translation System of `Kana' Representations to `Kanji-kana' Mixed Representations," *NHK Thechnical Journal* 138 Vol.25 No.5, 1973.
2. Brill, E. (1992) "A Simple Rule-Based Part of Speech Tagger," *Proceedings of the Fifth DARPA Workshop on Speech and Natural Language*, Morgan Kaufmann Publishers, San Mateo, CA. February 1992, pp. 112-116.
3. Church, K. A (1988), "Stochastic Parts Program and Noun Phrase Parser for Unrestricted Text," *Proceedings of the Second Conference on Applied Natural Language Processing, ACL,* 1988, 136-143.
4. de Marcken, C.G. (1990) "Parsing the LOB Corpus," *Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics* 1990, 243-251.
5. DeRose, S.J. (1988) "Grammatical Category Disambiguation by Statistical Optimization," *Computational Linguistics* 14: 31-39, 1988.
6. Meteer, M., Schwartz, R., and Weischedel, R. (1991) "Empirical Studies in Part of Speech Labelling," *Proceedings of the Fourth DARPA Workshop on Speech and Natural Language,* Morgan Kaufmann Publishers, San Mateo, CA. February 1991, pp. 331-336.
7. Weischedel, R. (1991) "A New Approach to Text Understanding," *Proceedings of the Fourth DARPA Workshop on Speech and Natural Language,* Morgan Kaufmann Publishers, San Mateo, CA. February 1991, pp. 316-322.

APPENDIX

To estimate the reliability of hypothesized correction rules, we used the sample inspection method. If the sample size is small, high frequency cases may tend to receive a higher probability estimate than if the sample were larger.

The sample inspection method provides an objective measure of how likely estimation error may be, given small samples. Suppose we have:

- a total of N elements in a population,
- R elements in a desired class,
- n sample elements in total, and
- r sample elements in the desired class

The conditional probaiblity of R > R1, given r = r1 will be:

$$p(R>R_1 | r=r_1) = \frac{p(R>R_1,\ r=r_1)}{p(r=r_1)}$$

Since we assume the elements of R occur independently, we have

$$= \frac{\sum_{R>R_1} p(R)\, p(r=r_1|R)}{\sum_{R>0} p(R)\, p(r=r_1|R)}$$

Assuming p(R) is approximately constant, we have

$$= \sum_{R>R_1} p(r=r_1 | R) \qquad (1)$$

Here p(r = r1 | R), the conditional probability of r desired elements given R desired elements in the population, is given by a hypergeometric distribution. The distribution will approach a binomial distribution as N gets larger.

$$p(r=r1 \mid R) = \frac{\binom{R}{r}\binom{N-R}{n-r}}{\binom{N}{n}} \qquad (2)$$

$$N \to \infty \text{ ------->} \binom{n}{r} q^r (1-q)^{n-r}$$

Therefore, substituting (2) to (1), given a significance level k (the probability that the conclusion is correct; for eacmple 0.9), we search for the largest q' which satisfies:

$$p(q>q' | r=r_1)$$

$$= \int_{q'}^{1} \binom{n}{r} q^r (1-q)^{n-r} > k$$