The Mega-Word Tagged-Corpus Project

Hiroshi Maruyama Shiho Ogino
IBM Research, Tokyo Research Laboratory
1623-14, Shimotsuruma, Yamato-shi, Kanagawa-ken 242, JAPAN
{maruyama,shiho,txhidano}@trl.vnet.ibm.com
Masaru Hidano
Toyohashi University of Technology
hidano@saylgw.tutics.tut.ac.jp

Abstract

Large corpora with part-of-speech tagging play a very important role in recent statistics-based and example-based natural language processing systems. However, no such corpora have become widely available for Japanese so far. Because the Japanese language has no explicit word boundaries, it is impossible even to count words without a corpus that has at. least word segmentations. This paper describes our attempts to develop a tagged corpus with over one million words taken from Japanese newspaper articles in a semi-mechanized way taken from Japanese newspaper articles. After dividing the original text into many chunks, we analyze the first chunk by using a Japanese morphological analyzer, and correct. the output manually; then, using that result, we improve the morphological analyzer and go on to the next chunk. Thus, the quality of the morphological analyzer increases at each iteration, decreasing the effort required for manual editing of the following chunks. Our experience in the first iteration of this 'boot-strapping' process has been encouraging.

1 Introduction

Recently, corpus-based approaches to natural language processing have drawn much attention. For example, at TMI-92, held in Montreal last year, 13 out of 23 papers considered some form of corpus as an essential source of linguistic information. Thus, it is becoming more and more important to have high-quality, large corpora.

It seems that corpora can be categorized according to the amount of preparation needed for the raw text.

A *word-segmented corpus* is simply a sequence of words, and can be used to extract such information as word frequency, word n-grams [2], and word cooccurrence counts [5]. Little preparation, if any, is needed to build this type of corpus for languages with explicit word boundary characters (that is, blanks) such as English.

In a *part-of-speech tagged corpus,* each word is associated with part-of-speech information. The famous Brown Corpus [6] and LOB Corpus are of this type. Building them manually must have taken a lot of effort, but they have already contributed significantly to natural language processing research such as Church's part-of-speech tagging program [3], and Su's ArchTran machine translation system [14]. Thus, the labor required for building a tagged corpus seems to be justified by the high-quality linguistic knowledge that can be derived from the tagged corpus.

Large *parsed corpora* are ideal for extracting knowledge, but they are rare. Most corpus-based research so far has been based on either a word-segmented corpus or a part-of-speech tagged

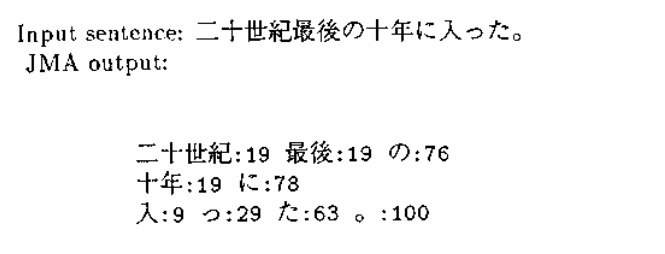

Figure 1: Sample input/output of JMA

corpus.

It is obvious that even word-segmented corpora are very useful for extracting various kinds of linguistic knowledge if they are sufficiently large, but unfortunately, it is not easy to build a Japanese word-segmented corpus for the simple reason that the Japanese language has no explicit word boundaries. We think this is the biggest barrier that prevents us from conducting a large-scale experiment using a corpus-based technique. Because of this difficulty, we initiated the current project. Our goal is to develop a high-quality tagged corpus of Japanese text that contains more than one million words. Once this corpus has been developed, many ideas that have proved to be very effective for English text can be tested on the Japanese language as well.

Because of our severely limited human resources, it is not possible to do all the development work manually. However. Japanese morphological analysis is a relatively well-established technology and many Japanese morphological analyzers have more than 95% accuracy. Therefore. it might be possible to minimize the human effort if we take machine analyses and post-edit them by hand.

This article describes the plan of our project and presents several insights that we have acquired so far. The next section gives a. description of our Japanese morphological analyzer. The process of building the tagged corpus, including both mechanized processes and human processes, is described in Section 3.

2 Japanese Morphological Analyzer

A Japanese morphological analyzer (hereafter called the JMA) takes an input sentence and segments it into words and phrases, attaching a part-of-speech code to each word at the same time. Figure 1 shows sample input and output of our JMA.

The grammatically of a sequence of Japanese words is mainly determined by looking at two consecutive words at a time (that is, by looking at two-word windows). Therefore, Japanese morphological analysis is normally done by using a Regular Grammar (e.g., Hisamitsu and Nitta [10]). Our JMA grammar rules have the following general form:

statel → "word" linguistic-features state2 cost.

Each grammar rule has a heuristic cost, and the parse with the minimum cost will be selected as the most plausible morphological reading of the input sentence. A part of our actual grammar is shown in Figure 2. Currently our grammar has about 4,300 rules and 400 nonterminal symbols.

2.0.1 Counting Errors

There are two types of error in Japanese morphological analysis. One is wrong identification of word boundaries. For example, the string "ABC" may be analyzed as two words. "AB" and "C," when the correct analysis is "A" and "BC." This type of error has a major impact on

```
文節頭 -> KAGYOU-5DAN [pos=1,kow=doushi] カ行５段;
カ行５段 -> "か" [pos=26,kow=v_infl] 動詞５段未然ナイ接続 cost=300;
動詞５段未然ナイ接続 -> "ず" [pos=64,kow=jodoushi,fe={negative}]
                        "に" [pos=78,kow=setsuzoku_joshi]
                        接続助詞ガ cost=500;
接続助詞ガ -> 文節尾 cost=999;
名詞 -> "" [pos=48,kow=jodoushi] 助動詞ダ語幹 cost=500;
助動詞ダ語幹 -> "だろ" [pos=27,kow=aux_infl] 未然ウ接続 cost=300;
助動詞ダ語幹 -> "だっ" [pos=45,kow=aux_infl] 連用タ cost=300;
```

Figure 2: Some of the grammar rules

content word								
noun	verb	adj.	adv.	attr.	affix	numeric	conj. etc.	total
13	17	4	6	1	4	1	2	48

function word				
inflection	particle	auxiliary stem	others	total
41	24	4	2	71

Figure 3: POS system

the quality of analysis, because it degrades the reliability of the statistics of word counts, word cooccurrence, and so on.

The other type of error is wrong association of a part-of-speech (POS) with words. The error rate of wrong POS depends on how detailed POS information is required to be; if a very coarse POS system is used, the chance of errors occurring can be made small. Our POS system has 119 different codes, which have been carefully selected to make them detailed enough for many applications while at the same time minimizing the error rate. Figure 3 gives a rough idea of the level of coarseness of our POS system.

We intend to collect statistical data about errors in terms of the number of correctly analyzed words, but counting errors requires a little more definition, because errors are sometimes interrelated, and one problem (e.g. a word missing in the lexicon) may cause wrong analyses of the preceding and succeeding words. For example, the string "AABBCC"' consisting of three words, "AA," "BB," and "CC," might be analyzed as "AAB"' and "BCC," because the word "BB" is missing in the lexicon. We take the most strict position: for every 'correct' word, if that word is not in the analysis, we assume that there is one error. In the above case, there are three words that are not identified correctly, so we count three errors.

The error rate of our JMA in analyzing the first 1,016 sentences (29,024 words) of the text corpus is shown in Table 1.

3 Building a Large Tagged Corpus

We have two contradicting goals: to keep the quality as high as possible and to minimize the human efforts required.

The importance of maintaining good quality should not be underestimated. If linguistic

Category	number	error rate
segmentation error	364	1.25%
POS error	323	1.11%
total	687	2.36%

Table 1: JMA error rate

knowledge extracted from low-quality data us used, only low-quality output can be expected.

In a previous paper, [12], we applied to the JMA an unsupervised stochastic training method (that is, one without human editing of the training corpus) [1] similar to that used in Fujisaki et al. [8], and found that it did not greatly improve the quality of analysis; in fact, it sometimes decreased it. This is because, in unsupervised training, some particular sequence of words is always analyzed in the same way, depending on the initial parameters. Thus, if the sequence is analyzed wrongly at the first iteration of the training, it will never be recovered unless there are sufficient evidence in the remaining corpus showing otherwise. Our conclusion from this experiment is that, even though unsupervised training works very well as a first approximation for relatively difficult tasks (such as machine translation [2]), it. is of little help in areas where conventional methods already attained a very high level of accuracy, such as Japanese morphological analysis.

Besides, since revising an existing tagged corpus to improve the quality takes nearly the same amount of time as its original development, it is a good idea to have a corpus of as high quality as possible from the beginning. Involving human in the post-editing costs is very costly, and we should avoid anything that might degrade the value (that is, the quality) of the corpus.

If the JMA produced 100% correct analysis, we would not need post-editing. Unfortunately this is not the case, and therefore we have to go through a. file like the one shown in Figure 1 and correct errors when we find them. High-quality JMA output may reduce this editing effort. Since the partially-built tagged corpus is an excellent source of linguistic knowledge that, can be used to improve the JMA itself, we can expect a significant reduction in the amount of post-editing required if we feed back information on the errors that, have occurred during previous Japanese morphological analysis processes to the subsequent processes. This idea of 'bootstrapping' is common in corpus-based approaches (e.g.. Nagao [13]), and we will explain the details of the process in the next subsection.

3.1 Bootstrapping

Since the partially developed tagged corpus can be used to improve JMA, once we have analyzed one chunk of data, we allow feedback be incorporated into the JMA for the next, chunk. For this purpose, we first divide the original text into many chunks.

Our text is taken from newspaper articles and has a total size of about two million characters. We divided it into 36 chunks called chunk#l, chunk#2. and so on. One chunk consists of approximately 1,000 sentences, or 60,000 characters. We first analyze chunk#l by using the JMA, edit the output manually to obtain 'corrected' data, and improve the dictionary and the grammar of the JMA for the rest of the chunks.

Figure 4 shows the process flow for one chunk. First, the input data are analyzed by the JMA (pass 1). The results of this analysis are stored in on disks (marked as "data of pass 1"). Looking at these data, the human editor checks for errors and corrects them by using a text editor. The product of this labor-intensive task is the 'corrected' version of the analysis.

During this task, we found that query-and-replace-type software tools implemented as an editor macro were very useful. For example, our Chunk#l contains many articles about the

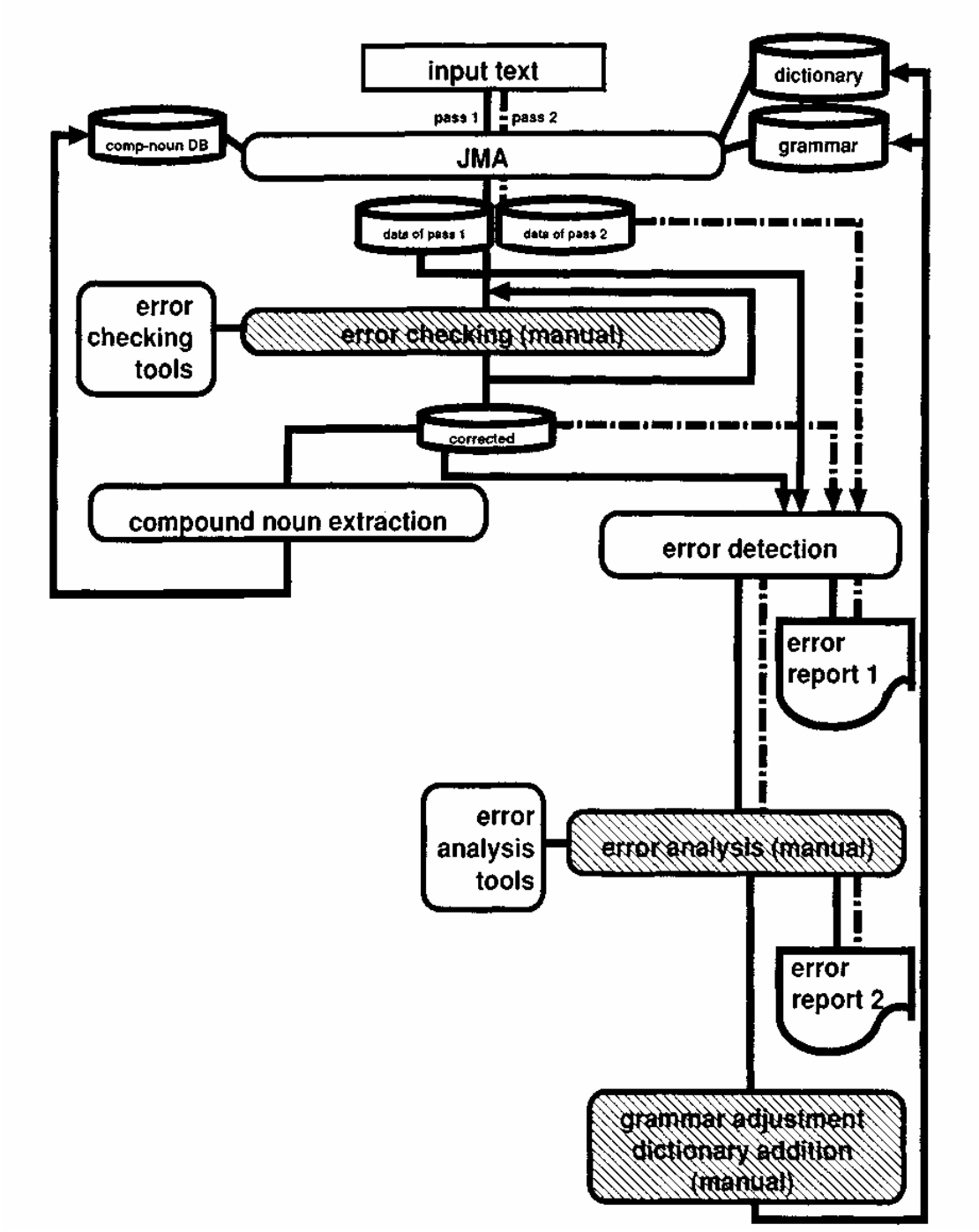

Figure 4: Process flow

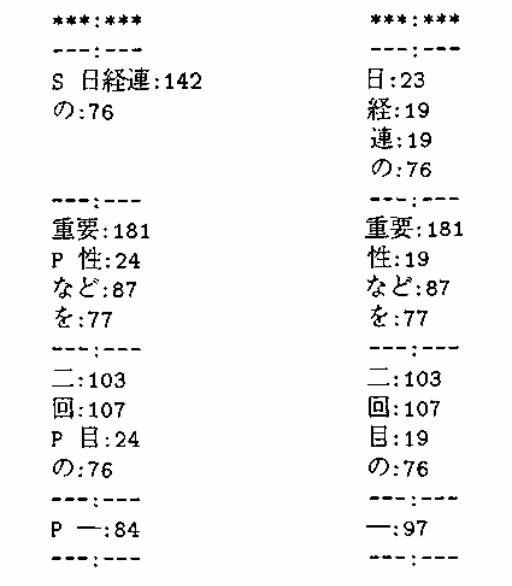

Figure 5: Error report 1

Gulf War, and since our original dictionary does not contain the entry "Wangan'" (Gulf), every occurrence of "Wangan" was wrongly analyzed as two words. When we encountered one such occurrence, we could anticipate that the same error might occur many times in the rest of the data, *so* we initiated a query-and-replace macro to make a collective change.

This 'corrected' version is compared with the original analysis, and error report 1 is generated automatically from the difference between these two files. A part of the error report 1 generated for the first chunk of input text is shown in Figure 5. In the figure, the left column shows the correct parsing of phrases, while the right column shows the erroneous analysis. The error code (either 'S' or 'P' attached to the word in the left column) indicates the error category defined in Section 2 ('S' stands for segmentation error and 'P' stands for POS error).

3.2 Improvement of the JMA

To improve the JMA, we first analyze the causes of the errors listed in error report 1. Unfortunately we cannot mechanize this process, because an error may have various causes such as

- a missing entry in the lexicon.
- a wrong or missing part-of-speech in the lexicon.
- a missing rule in the grammar.
- a wrong part-of-speech in the grammar, or
- an inappropriate setting of costs in the grammar,

and sometimes these problems are combined and/or interrelated. In addition, proper disambiguation is sometimes impossible without context information. In such cases, the current

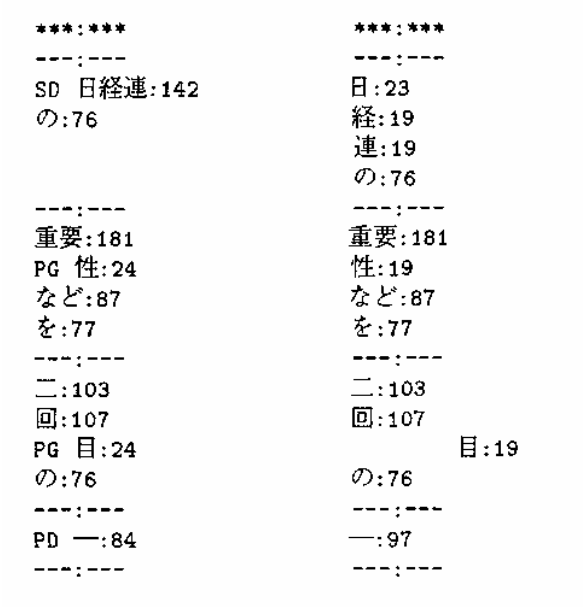

Figure 6: Error report, 2

linguistic model (that is, Regular Grammar) cannot produce a correct analysis even if a perfect lexicon and a perfect grammar are provided.

Therefore, the 'error analysis' task in the figure requires a skilled person who understands the grammar, the lexicon, and the algorithm very well, in contrast to the 'error checking' task. which can be done by any native speaker with a little training. The product of the 'error analysis' task is a list of the problems that caused the errors shown in error report 1. This is called error report 2. An example of error report 2 is shown in Figure 6. In addition to the 'S' and the 'P' flags in error report 1, error report 2 also contains the *reason code* 'D' (dictionary content error) or 'G' (grammar error).

One tool that has proved to be useful during this 'error analysis' task is a. dictionary-lookup tool. If a word listed in error report 1 does not appear in any lexicon, especially when it is a, content word, it is very likely that the problem lies with the lexicon.[1]

According to error report 2, a plan for modifying the grammar and the lexicon is made, and is then implemented in the JMA.

In addition to the above modifications of the grammar and the dictionary, there is a, third way to improve the accuracy of the JMA. It, is related to the analysis of compound nouns, and uses a database of compound nouns that are included in the analyzed text. Wrong segmentations of compound nouns accounts for about half of all word-segmentation errors. On the other hand, correct segmentations of compound nouns are accumulated during iterations over the chunks. Therefore, we decided to build a compound-noun database automatically from the corrected data. When the same compound nouns appear in the subsequent, analysis, the JMA uses the result found in the database instead of analyzing the internal structure of the compound noun. In the 1,016 sentences in the first chunk, 1,361 compound nouns were found and placed in the database.

[1] All the function words and some content words are in the grammar rather than the lexicon, because they need special treatment in analysis. Therefore, the absence of an entry in the lexicon does not necessarily indicate a dictionary problem.

grammar	compound noun	dictionary	error rate
no	no	no	2.36%
yes	no	no	2.58%
yes	yes	no	1.89%
yes	yes	yes	1.39%

Table 2: Effects of JMA improvement

3.3 Evaluation of Improvement

After the JMA has been modified, (the grammar, the lexicon, and the compound-noun database), the effect of the modification is measured. This is clone fully automatically by applying the JMA to the same input text again (pass 2 in Figure 4) and comparing the result with the 'corrected' data. The effects of the three kinds of modification that are made in our first iteration are shown in Table 2.

4 Conclusion

We started the current project in January 1993, and expect it- to be finished by the end of 1994. During and after this period, we believe that we will come up with a lot of new techniques and ideas that can be tested by using our high-quality linguistic data.

References

[1] Baum, L. E., "An Inequality and Associated Maximization Technique in Statistical Estimation for Probabilistic Functions of Markov Processes," *Inequalities,* Vol. 3, 1972.

[2] Brown, P., Cocke, J., Pietra, S. D., Pietra, V. J. D., Jelineck, F., Lafferty, J. D., Mercer, R. L., and Roossin, P. S., "A Statistical Approach to Machine Translation," *Computational Linguistics,* Vol. 16, No. 2, 1990.

[3] Church, K. W., "A Stochastic Parts Program and Noun Phrase Parser for Unrestricted Text," *Proc. of ACL 2nd Conference on Applied Natural Language Processing,* Austin. Texas, 1988.

[4] Church, K. W., Gale, W., Hanks, P., and Hindle, D., "Parsing, Word Associations and Typical Predicate-Argument Relations," *Proc. of International Parsing Workshop* '89, Pittsburgh, 1989.

[5] Church, K. W. and Hanks, P., "Word Association Norms, Mutual Information, and Lexicography," *Computational Linguistics,* Vol. 16, No. 1, 1990.

[6] Francis, W. and Kucera, H., *Frequency Analysis of English. Usage,* Houghton Mifflin Company, Boston, 1982

[7] Fu, K. S., *Syntactic Methods in Pattern Recognition,* Academic Press, 1974,

[8] Fujisaki, T., Jelinek, F., Cocke, J., and Black, E., "Probabilistic Parsing Method for Sentence Disambiguation," *Proc. of International Parsing Workshop '89,* Pittsburgh, 1989.

[9] Jelinek, F. et al., "Continuous Speech Recognition by Statistical Methods," *Proc. of the IEEE,* Vol. 64, No. 4, 1976.

[10] Hisamitsu, T. and Nitta, Y., "A Uniform Treatment of Heuristic Methods for Morphological Analysis of Written Japanese," *Proc. of 2nd Japan-Australia Joint Workshop on NLP,* 1991.

[11] Kuno, S., "The Augmented Predictive Analyzer for Context-Free Languages and Its Relative Efficiency," *Comm. ACM,* Vol. 9, No. 11, 1966.

[12] Maruyama, H., Ogino, S., and Watanabe, H., "Stochastic Japanese Morphological Analysis," (in Japanese) *Proc. of 8th Annual Meeting of Japan Society for Software Science and Technology,* Sapporo, 1991.

[13] Nagao, K., "Dependency Analyzer: A Knowledge-Based Approach to Structural Disambiguation," *Proc. of COLING '90,* Helsinki, 1990.

[14] Su, K.-Y., Wu, M.-W., and Chang, J.-S., "A New Quantitative Quality Measure for Machine Translation Systems," *Proc. of COLING '92,* Nantes, 1992.