Yuji MATSUMOTO, Nara Institute of Science and Technology,
Takenobu TOKUNAGA, Tokyo Institute of Technology
Manabu OKUMURA, Japan Advanced Institute of Science and Technology
Masaharu OBAYASHI, Kanrikogaku Ltd.

A Computational Lexicographer's Workbench

Abstract

The paper presents a computational environment to support developing a lexicon for natural language processing. The environment provides four independent but closely related processing modules: language analyzers (morphological and syntactic dependency analyzers) used for accumulation of tagged corpus, a searcher and a browser of tagged corpora, a lexicon compiler, and language resource exchangers. The underlying idea is to utilize up-to-date language technologies to minimize both the human labour and the inconsistency that are unavoidable in manual compilation of a lexicon. The proposed computational environment enables an efficient construction of a consistent and fertile lexicon. The target language of the project is Japanese.

Keywords: statistical language processing, computational lexicography, corpus, lexical acquisition

1. Introduction

It is indispensable to have a large scale lexicon (dictionary) for practical natural language processing. There are, however, notorious problems that originate in the diversity of language use. Even in a single language, the usage of words varies according to the fields and the purposes. Existing Machine Readable Dictionaries (MRDs) are, in many cases, not satisfactory for natural language processing since they are usually designed for general purposes, do not enumerate possible usages in specialized areas, and are unable to catch up with novel word usages. Corpora (or text databases) are valuable objective resource that possibly compensates those deficits and enhances existing MRDs to compile a computational lexicon for practical natural language processing.

This paper presents a current achievement of our project aiming at construction of an environment for compilation of a computational lexicon. Since the language we use in the project is Japanese, most of the tools are designed especially for Japanese. However, the design concepts, basic tools and interfaces are language independent. The whole environment is called Computational Lexicographer's Workbench (CLW). A lexicon for natural language processing should include a large scale vocabulary, syntactic and semantic information of words with examples that show their usages in a systematic way. The workbench provides statistical language processors such as a part-of-speech (POS) tagger, a syntactic analyzer (a parser) for dependency analysis, a pattern-matcher and a browser for tagged (POS-tagged or parsed) corpora.

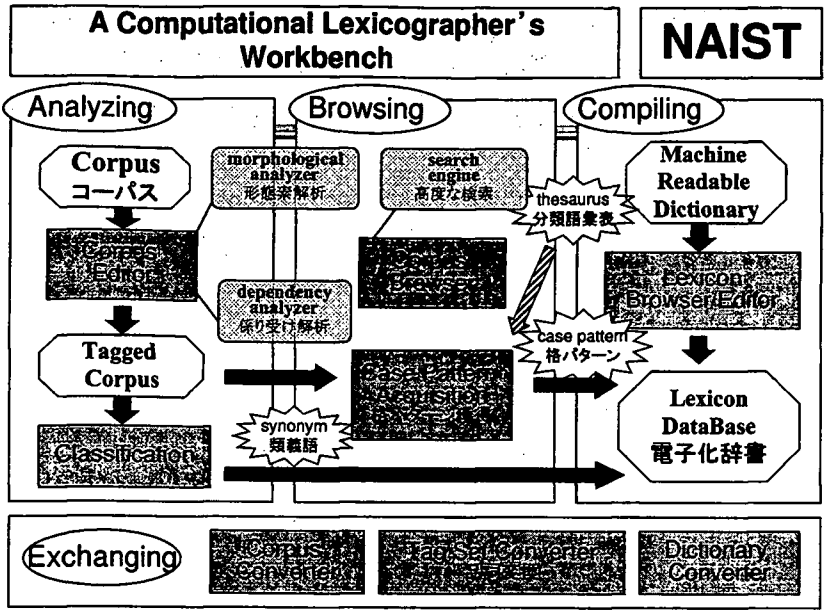

Figure 1: Overview of Computational Lexicographer's Workbench

The language processors are based on statistical models that learn their probabilistic parameters from tagged corpora. Those systems are used as the pre-processors for untagged corpora that are to be corrected by human inspectors. The cycle of automatic corpus tagging, tagging-error correction and probabilistic parameter learning for the language processors forms an effective procedure of accumulation of correctly tagged corpora. The accumulated corpora serve as the base resource for lexical information extraction. The overall conception is to minimize both human labour and the inconsistency that may be caused by manual compilation of a lexicon.

We should emphasize that the main concern of the project is actually not to build a lexicon. The real objective is to build an environment that is supported by language processing facilities to construct, modify, transform, and customize a computational lexicon.

2. Overview of Computational Lexicographer's Workbench

The overview of the computational lexicographer's workbench is shown in Figure 1. The workbench consists of four major components, which are summarized as follows:

- **Language Analysis Component** provides a morphological analyzer (part-of-speech tagger) and a syntactic dependency analyzer. Both analyzers are statistic based systems, in which probabilistic parameters are learned from tagged corpora to improve the accuracy of the systems. Analyzed sentences are accumulated in SGML format and form a repository of tagged/bracketed corpora.
- **Browsing Component** provides a pattern-matcher and a browser to be used for tagged corpora. The pattern-matcher operates with regular expressions that take into account not only lexical forms and parts-of-speech tags but also syntactic dependency

relations. This component is used to collect specified usages of verbs, from which their case frame patterns are extracted.

- **Lexicon Compilation Component** provides the facilities of compilation and modification of partially constructed lexicon. Other important facilities of this component are retrieval of lexical entries, consistency checking of lexical descriptions, and a hyperlink facility for flexible access to the lexicon.
- **Resource Exchange Component** supports conversion of tagged corpora or a lexicon in one tag set into that of another tag set. This component is necessary since a number of distinct tag sets are used in different research groups and it is important to share natural language resources even with such different communities.

Figure 1 gives an overview of those components and their relationship. The first three modules are used in an integrated way for lexicon compilation. Each component and each facility is usable as an independent natural language processing tool. The resource converters are used to transform a tagged corpus or a lexicon into that of another tag set or dictionary format. The following sections describe those components in detail.

3. Language Analysis Component

This component provides the tools for corpus analysis, including a morphological analyzer (POS tagger) and a syntactic dependency parser. They are statistics based systems that learn their probabilistic parameters from correctly tagged corpora. The morphological analyzer, called ChaSen (Matsumoto et al. 1997), is a cost-based part-of-speech tagger for Japanese. A cost is defined for each token and for each pair of consecutively appearing POS tags. The system determines the sequence of POS tags for the input sentence that gives the least sum of the costs. The search is executed in the time proportional to the length of the input sentence. It uses a dynamic programming algorithm. The original algorithm is equivalent to the POS tagger based on the bi-gram Hidden Markov Model (cf. Charniak 1993) and the parameters (costs) are learned from a tagged corpus. The inverse logarithm of the probability values are used as the cost so that multiplication of the probability values are replaced with summation of the costs. The dependency parser is also based on a statistical method that measures the probabilities of dependency relation between a pair of lexical items. The basic algorithm is similar to Collins' probabilistic dependency parser (Collins 1996) and is a modification to it (Fujio and Matsumoto 1998). Both systems are equipped with an interface for showing and correcting the tagged results. A human instructor confirms or modifies the output of the systems through simple mouse operations. In this paper, we simply use the word *tagged* to refer either to a POS tagged corpus or to syntactic tagged (bracketed) corpus.

By accumulating tagged corpora, the system performance is enhanced through statistical learning procedures. The original bi-gram model has been replaced with variable length Markov Model (Ron et al. 1997), in which the length of the contexts used to identify ambiguous POS tags is changeable as necessary and is determined on the basis of statistical importance (Haruno and Matsumoto 1997). The accumulated tagged corpora also serve as the base resource for the lexicon compilation. This achieves an effective cycle of acquisition of tagged corpora and of enhancement of language processors. The tagged sentences are stored in an SGML format. Two major SGML tags for parsed corpus are those of words and phrases. A word tag specifies the reading, the POS tag and the base (stem) form of the word. A phrase tag specifies the range of a small phrase (a Japanese phrasal fragment called a *bunsetsu*), its

head word (the semantical head word), the name of dependency relation (the modification type) and the identifier of the phrase it modifies. The interface for syntactical dependency parsing of sentences is shown in Figure 2. Each small box in the figure corresponds to a small phrase (bunsetsu) and an edge connects a modifier and its modifiee. The edges to the right of the boxes specify the syntactic dependencies determined by the dependency analyzer, while the edges to the left are the ones corrected by human inspection. Corrections on the contents of the small phrases as well as the POS tags can be done through this interface.

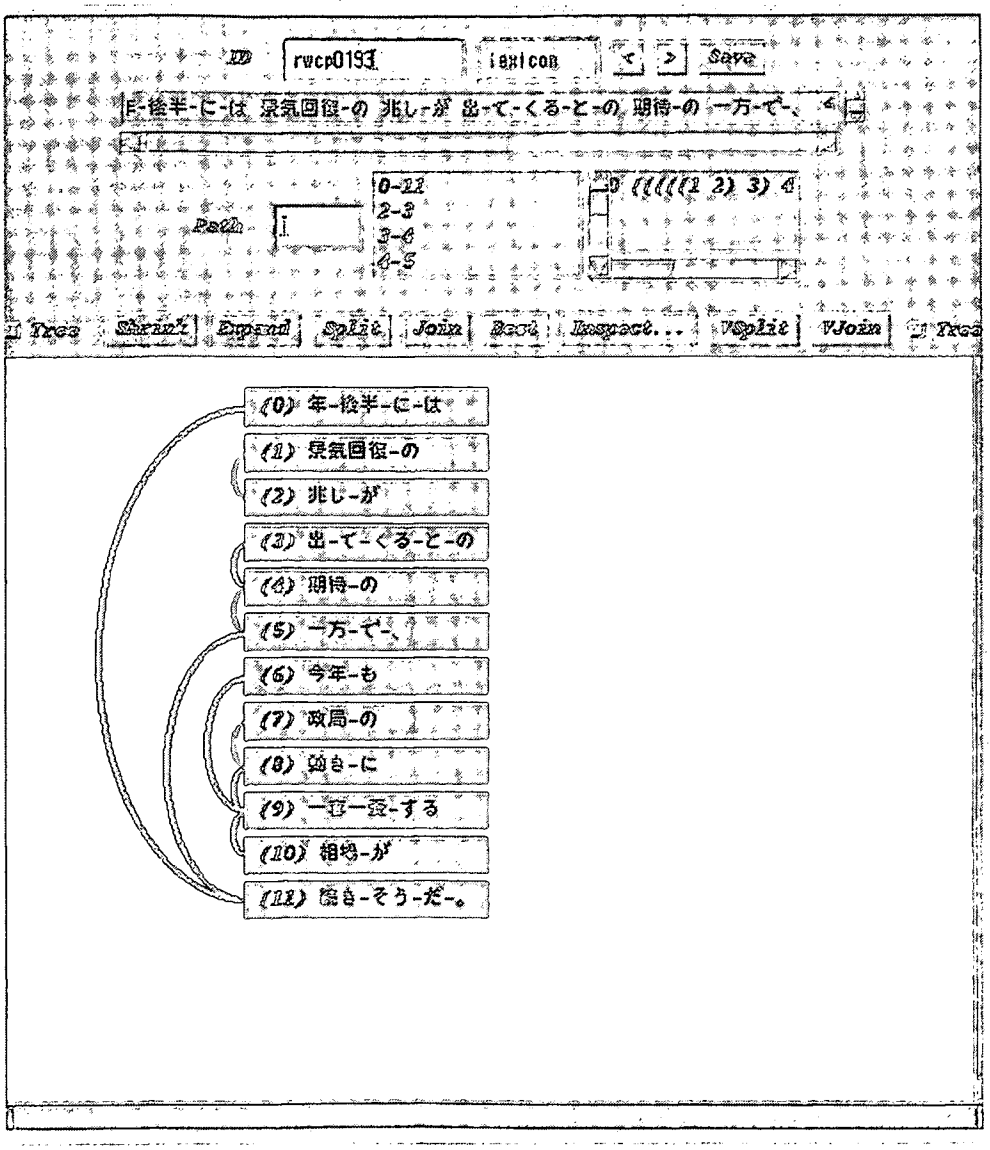

Figure 2: Interface for Syntactic Dependency Analyzer

Semantic similarity between lexical items is useful information. A similarity measure is defined from the collocation between nouns and verbs, and is applied to construction of thesauri by using a hierarchical clustering algorithm (Tokunaga et al. 1995).

4. Browsing Component

```
<SENTENCE ID=''tst0'' USER=''lexicon'' STAT=''depend''>
<SEGMENT HEAD=''2'' REL=''が'' DEP=''2''>
<W READ='' タロウ'' BASE='' 太郎'' POS='' 名詞-固有名詞-人名-名''>太郎</W>
<W READ='' ガ'' BASE='' が'' POS='' 名詞-一般''>が</W>
</SEGMENT>
<SEGMENT HEAD=''2'' REL=''を'' DEP=''1''>
<W READ='' ロウカ'' BASE='' 廊下'' POS='' 名詞-一般''>廊下</W>
<W READ='' ヲ'' BASE='' を'' POS='' 助詞-格助詞-一般''>を</W>
</SEGMENT>
<SEGMENT HEAD=''0'' REL=''/動詞-自立/基本''>
<W READ='' ハシル'' BASE='' 走る'' POS='' 動詞-自立'' CTYPE='' 五段・ラ行'' CFORM=''
基本形''>走る</W>
</SEGMENT>
</SENTENCE>
```

Figure 3: Parsed sentence in SGML format

This component provides a browsing facility of tagged corpus. The search engine accepts a retrieval query in a form of regular expression composed of lexical items, POS tags and syntactic dependency relations. Figure 3 shows a sample of a parsed sentence in the SGML format. The arguments of the 'SENTENCE' tag is the sentence identifier (ID), the human inspector's user id (USER), and the level of analysis (depend indicates that the sentence is analyzed at the syntactic dependency level). 'SEGMENT' means a Japanese small phrase (*bunsetsu*), which forms the basic unit of dependency structures. This tag contains the information of the head (the main word) of the segment as well as the segment number to which it modifies. The segment number is allocated from the beginning of the sentence starting from 0. The word tag 'W' contains the reading, the base form and the POS tag of the word.

Each constituent in a query may be specified in any piece of those descriptions. For example, a query can specify a regular expression composed of surface word forms or of partially specified part-of-speech tags. The part-of-speech tag set is defined hierarchically, in that noun is classified into common noun, proper noun, verbal noun, dependent noun, suffix noun, and so on. Furthermore, proper noun is classified into personal name, place name, and organization name, which are further classified into more fine-grained part-of-speech tags. The current tag set is classified into a four-level hierarchy. An interesting and complicated query can be issued to the system, such as "a noun phrase headed by an organization name that modifies a verb 'invest' as a subject." The system is accompanied with an interface that displays the retrieved data in a KWIC (key words in context) format. The user can describe the format in which the results are displayed, e.g., which constituent is aligned as the center element, which information is used to sort the results, which colour is used to highlight the matching patterns in the retrieved data, and so on.

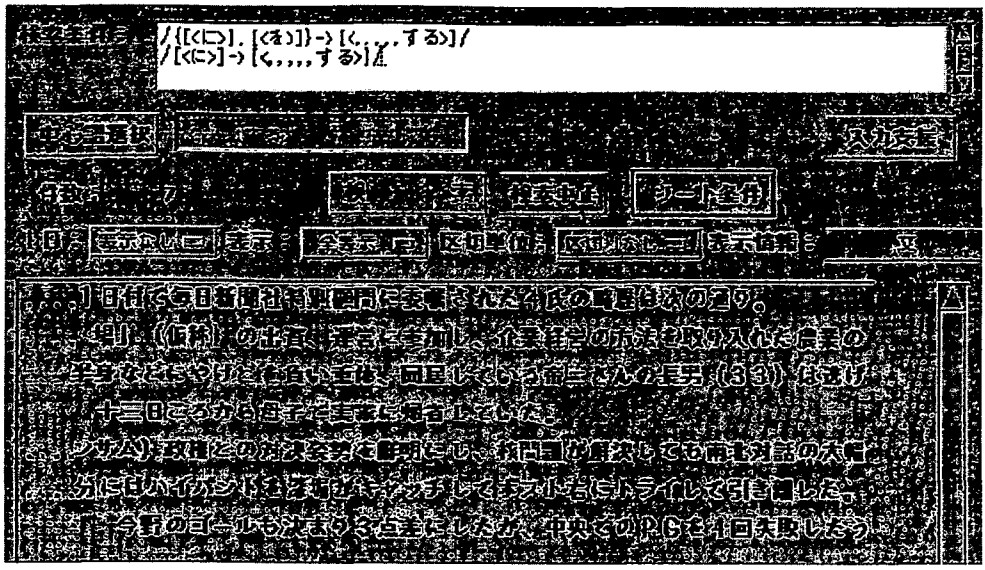

Figure 4: Corpus Browser

Figure 4 shows a part of pattern matching results of the query pattern, "a case particle *ni* modifies a verb *suru (do)*." Since Japanese verb *suru* has various inflected forms that are mutually quite different, it is difficult to identify the occurrence in texts without performing morphological analysis. Different forms of *suru* is aligned as the center element of the KWIC in Figure 4.

Once a set of analyzed data for a verb are collected using the system, the Case-pattern Acquisition system makes up a preliminary subcategorization frame(s) for a verb with selectional restriction described by the appropriate semantic classes in an existing thesaurus (Utsuro and Matsumoto 1997). The automatically constructed thesaurus (Tokunaga et al. 1995) can also be used for this purpose.

5. Lexicon Compilation Component

Compilation and maintenance of a lexicon requires a large amount of human labour, and therefore, the environment to help human compilation and maintenance of a lexicon is indispensable.

This component consists of supporting tools for lexicographers to compile a lexicon. Two base resources are assumed: Existing MRDs and the preliminary lexical description constructed through the previous components. The lexicon compilation component provides database facilities to define and constrain the form of lexical entries. Initial description of lexical entries are obtained from the above resources or from scratch ensuring to satisfy the constraints expressed by the lexicon database definition. By virtue of the constraints and the objective use of language resources, this module helps a compilation of a consistent and fertile lexicon.

The system provides a hypertext facility where some of the hyperlinks are automatically constructed from a machine readable dictionary. The process of hypertext construction from a dictionary plays an important role in checking inconsistency among description of entries in the dictionary. In trying to connect two entries by a link, the inconsistency between the description can be automatically detected.

The hypertext representation of a lexicon also provides another important function as an interface for compiling and maintaining a dictionary. The main characteristic of a hypertext is its nonlinear network of interconnection between entries. This enables an easy movement among multiple entries of interrelated lexical items.

This function is useful not only for lexicographers, but also for users of the lexicon. Browsing of a hypertext lexicon is a powerful way of lexicon access, especially in case users are unfamiliar with the structure of the lexicon.

The facilities provided by this component are summarized as follows:

- Format of lexical entries defined for each part-of-speech
- Conversion facility from existing MDR format to user-defined format
- Consistency checking of description of lexical items
- Graphical user interface and hyperlink facility to browse and to compile lexical items

6. Resource Exchange Component

This component provides a set of resource conversion tools: The Tag Set Converter compares two tagged corpora that are analyzed based upon different tag sets, helps the user to identify the correspondence between the tag sets, and finally derives transformation rules between the tag sets. The Corpus Converter applies the transformation rules to convert a tagged corpus with a tag set into the one with the other tag set. The Corpus Converter is equipped with a user interface to show the differences and the correspondences of the tag sets and prompts the user to make transformation rules by just selecting proposed rules or by modifying them. Since there are a number of distinct tag sets for Japanese lexicon proposed by different research groups or research communities, it is important to have such a facility for sharing linguistic resources. The dictionary Converter supports to transform a lexicon into another format.

7. Concluding Remarks

As stated in the introduction, the main concern of the project is not to build a particular lexicon. The aim is rather *to build an environment* to help building a lexicon for the user's own needs. Although the current systems commit to a specific tag set and a specific form of lexicon (IPA Lexicon (Kuwahata et al 1996)), the tools are designed flexible so as to adapt themselves easily to changes of grammatical definition. Moreover, each processing element can be used as an independent language processing tool.

Our project shares many aspects with ARIOSTO (Basili and Pazienza 1997). Both of the projects develop statistics based language processors to achieve an effective cycle of corpus analysis and system performance enhancement, and utilize corpora as the important resource for lexicon construction. This is a natural way to use corpus data for that purpose. Although the fundamental idea is similar, most of the components takes up distinct technique and there are novel facilities in our project, such as resource converters and search engine for regular expressions with syntactic dependency relation.

Acknowledgements

This project is supported in part by the Advanced Software Enrichment Project under the auspices of Information-technology Promotion Agency, Japan (IPA). The authors also appreciate discussion and cooperation with Minako Hasimoto of Fujitsu, Yoshiki Sugiura of Kanrikogaku Ltd., Kazuhisa Yamada of C-Labo and others who participated in various ways in this project.

8. References

Basili, R. and Pazienza, T. (1997). Lexical Acquisition and Information Extraction. *Information Extraction*, Lecture Notes in Artificial Intelligence 1299, Springer, pp.44-72.

Charniak, E. (1993). *Statistical Language Learning*. The MIT Press.

Collins, M. (1996). New Statistical Parser Based on Bigram Lexical Dependency. *Proc. 34th Annual Meeting of Association for Computational Linguistics*, pp.184-191.

Fujio, M. and Matsumoto, Y. (1998). Japanese Dependency Structure Analysis based on Lexicalized Statistics. *Proc. Third Conference on Empirical Methods in Natural Language Processing.*

Haruno, M. and Matsumoto, Y. (1997). Mistake-Driven Mixture for Hierarchical Tag Context Trees. *Proc. 35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics *Proceedings of the Conference*, Madrid, pp.230-237.

Kuwahata, W., Hasimoto, M. and Aoyama, F. (1996). Sense of Polysemous Nouns: Building a Computational Lexicon of Basic Japanese Nouns. *Proc. 16th International Conference on Computational Linguistics*, Copenhagen, Vol.2, pp.1082-1085.

Matsumoto, Y., et al. (1997). Japanese Morphological Analyzer ChaSen 1.0 Users Manual. *NAIST Technical Report*, NAIST-IS-TR97007, Nara Institute of Science and Technology, Nara, Japan.

Ron, D., Singer, Y. and Tishby, N. (1997). The Power of Amnesia: Learning Probabilistic Automata with Variable Memory Length. *Machine Learning special issue on COLT94.*

Tokunaga, T., Iwayama, M. and Tanaka, H. (1995). Automatic Thesaurus Construction Based on Grammatical Relation. *Proc. Fourteenth International Joint Conference on Artificial Intelligence*, pp.1308-1313.

Utsuro, T. and Matsumoto, Y. (1997). Learning Probabilistic Subcategorization Preference by Identifying Case Dependencies and Optimal Noun Class Generalization Level. *Proc. Fifth Conference on Applied Natural Language Processing*, pp.364-371.