A Case Frame Learning Method for Japanese Polysemous Verbs

Masahiko Haruno
NTT Communication Science Laboratories
1-2356 Take Yokosuka, Kanagawa, 238-03 Japan
E-mail: haruno@nttkb.ntt.jp

Abstract

This paper presents a new method for learning case frames of Japanese polysemous verbs from a roughly parsed corpus when given a semantic hierarchy for nouns (thesaurus). Japanese verbs usually have several meanings which take different case frames. Each contains different types and numbers of case particles (case marker) which in turn select different noun categories.

The proposed method employs a bottom-up covering technique to avoid combinatorial explosion of more than ten case particles in Japanese and more than 3000 semantic categories in our thesaurus. First, a sequence of case frame candidates is produced by generalizing training instances using the thesaurus. Then to select the most plausible frame, we introduce a new compression-based utility criteria which can uniformly compare candidates consisting of different structures. Finally, we remove the instances covered by the frame and iterate the procedure until the utility measure becomes less than a predefined threshold. This produces a set of case frames each corresponding to a single verb meaning. The proposed method is experimentally evaluated by typical polysemous verbs taken from one-year newspaper articles.

Introduction

Verbal case frames specify what combinations of a verb and phrases are acceptable in a sentence. They are applied in all practical aspects of natural language processing, (e.g., structural and semantic disambiguation, word selection in machine translation). It is therefore important to extract case frame dictionaries from large corpora. Several studies have been attempted to automatically acquire the co-occurrence relations from mutual information criteria [Hindle, 1990; Resnik, 1992; Grishman and Stering, 1992]. Their main target was **two dimensional** (binary) relation such as verb-object patterns in English. However, Japanese verbs subcategorize various numbers of phrases and are usually polysemous. Different meanings of a single verb take different types and numbers of case particles, each of which subcategorizes different noun categories. In addition, not all elements of a verbal case frame explicitly appear in a sentence since zero pronouns are often preferred in Japanese expressions. Hence, a learner needs to acquire a set of **multi dimensional** case frames. To meet these requirements, we devise a new learning technique featuring a **bottom-up covering method** [Michalski, 1983] and a new **compression-based utility criteria** of case frames.

The bottom-up covering approach is adopted to avoid the combinatorial explosion inherent in the exhaustive search method. Japanese has more than ten case particles and our thesaurus contains 2800 noun categories. It is clearly infeasible to enumerate all possible case frames and evaluate their utility against the corpus. Instead, by generalizing instances, it generates a sequence of case frame candidates, out of which the most plausible case frame is selected. Iterating the procedure generates several case frames each corresponding to a single verb meaning.

The compression-based utility measure is used in the case frame selection process above. It is based on the information complexity theory [Kolmogorov, 1965] which evaluates hypotheses by the amount of information (bits) needed to reproduce a set of examples. In other words, better hypotheses compress the examples more. The measure is highly suitable for our task because the utility is determined only by the compression power of a case frame and independent to the length of the frame unlike mutual information criteria. In addition, the compression-based model provides a well-founded background for multi dimensional case frame acquisition, enabling us to

- avoid over-fitting against the examples provided because it discriminates the examples to be encoded in a rule from those regarded as exceptions.
- acquire the optimally generalized case frames that provide the examples with the best explanation.

The first means that the proposed method is noise-tolerant and learns well even when only a small number of examples is provided. The second is a form of Occam's razor principle.

This paper is constructed as follows. After a brief presentation of our semantic hierarchy (thesaurus), we explain how to generate case frame candidates. Next, we discuss our compression-based utility measure and then present the overall bottom-up covering algorithm of our system. Finally, we confirm that the proposed method is remarkably useful by the experimental results for Japanese polysemous verbs taken from newspaper article.

Case Frame Candidate Generation

Semantic Hierarchy Used

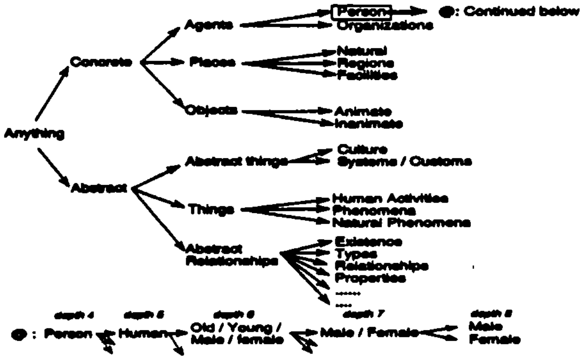

Figure 1: The Upper Levels of the Semantic Hierarchy

Figure 1 shows the structure of our semantic hierarchy [Ikehara *et al.*, 1993]. The hierarchy is a sort of concept thesaurus represented as a tree structure in which each node is called a category. An edge in this structure represents an "is-a" relation among the categories. For example, "AGENTS" and "PERSON" (see Figure 1) are both categories. The edge between these two categories indicates that any instance of "PERSON" is also an instance of "AGENTS". We implement the current semantic hierarchy as a set of Prolog (is-a) facts which is 12 levels deep and contains about 2800 nodes. Such level of detail was found necessary to perform semantic analysis enabling real world machine translation [Ikehara *et al.*, 1993].

Least General Generalization of Instances

Two examples below demonstrate how corpus sentences are transformed into training examples[1]. Only verbs and particle-noun pairs subcategorized by the verbs are considered, other constituents, e.g., adjectives, adverbs, relative clauses in the original sentence are ignored. This trimming is performed by a human supervisor with our treebank acquisition system. It suggests several possible syntactic structures for a sentence, from which the human supervisor selects the most plausible one. Outcomes are represented as a list of pairs consisting of case particle with dominant nouns.

(1) 彼が 突然 会社で 同僚と ビールを
kare-ga totuzen kaisha-de douryou-to bi-ru-wo
he-GA *suddenly office*-DB *colleague*-TO *beer*-WO
飲んだ
nomu+ta
drink+PAST

⇓

case([HEAD=>*drink*,
GA=>*he*, DB=>*office*, TO=>*colleague*, WO=>*beer*]).

(2) 昨日の 夕方に 近所の 子どもが
kinou-no yuugata-ni kinjo-no kodomo-ga
yesterday-NO *evening*-NI *neighbor*-NO *children*-GA
ワインを 飲んだ
wain-wo nomu+ta
wine-WO *drink*+PAST

⇓

case([HEAD=>*drink*, NI=>*evening*, GA=>*child*,
WO=>*wine*]).

A case frame is the information shared by a number of instances. It is therefore possible to generate case frame candidates by generalizing two instances. The **least general generalization (lgg)** of two instances is the procedure in which only the values of shared attributes are replaced by the least upper bound of the values using the thesaurus. It corresponds to a limited form of ψ term generalization [Ait-kaci and Nasr., 1986]. If the multiple inheritance of nouns causes several upper bounds of the same depth, we correspondingly produce several lggs. The following is the lgg of the above instances. HEAD, GA and WO are shared attributes and the least upper bounds of them (i.e., *drink-drink*, *he-child*, *beer-wine*) are *drink*, HUMAN and ALCOHOL, respectively. The notation of X: HUMAN symbolizes every noun assigned to category HUMAN in the thesaurus. *drink* in the lgg literally means the generalization of the verbs is *drink* itself.

(3) lgg case([HEAD=>*drink*, GA=>X:HUMAN, WO=>Y:ALCOHOL]).

Conversely, a case frame is expected to cover an example only if the case frame contains all attributes appearing in the example and values of the case frame are assigned to a category above those of the corresponding values of the example. Example (4) is covered by lgg (3) because *doctor* and *beer* are subconcepts of HUMAN and ALCOHOL, respectively.

(4) case([HEAD=>*drink*, GA=>*doctor*, WO=>*beer*]).

Compression-Based Case Frame Utility

The compression model used in this paper is related to the theory of algorithmic information theory [Kolmogorov, 1965]. The basic idea of the theory is that the complexity of any string *s* is taken to be the length

[1] Particles are explicitly marked in small caps.

(in bits) [2] of the shortest Universal Turing machine program required to generate s. Turing machine T has input and output tapes. The input tape comprises two components; the encoding of a case frame and the encoding of a derivational proof. The former encodes a case frame and the latter encodes a proof delineating how examples are reproduced using the case frame. The output tape literally enumerates the examples covered by the input case frame. A case frame is considered more plausible when there is more difference between the input tape length (in bits) and the output tape length (in bits). In other words, the more compactly a case frame compresses the examples, then the more plausible it is.

Case Frame Length

The easiest way to code a case frame is to transmit it directly in a sequence of characters. This, however, is extremely inefficient. According to information theory, the concept with probability p can be encoded in $-log_2 p$ bits at best. Under the assumption that all case particles and noun categories appear with the same probability, the following simple, but reasonably efficient encoding scheme can be used.

If there are P case articles in the target language and Cat noun categories in our thesaurus, in order to encode a case frame F of length N, we need

1. $log_2 N + log_2({}_P C_N)$ bits to encode case particles used
2. $N \times log_2(Cat)$ bits to encode the noun categories of each case

Adding these two together, case frame length defined as $CL(F)$ represents the compactness of F. Any increase in the length of F makes $CL(F)$ increase at the same time.

$$CL(F) =$$
$$log_2 N + log_2({}_P C_N) + N \times log_2(Cat)(bits)$$

Proof Length

The proof derives examples with respect to a case frame. Our proof encoding scheme makes use of the number of choice points in an SLD refutation using the standard Prolog leftmost computation rule as in [Muggleton *et al.*, 1992]. The number of choice points reflects proof complexity well since it represents nondeterminism at every step of the proof. To prove an example in terms of a case frame, we have to show that all nouns of the example are subcategories of correspondents of the case frame by tracing down a thesaurus. Thus, the number of choice points in the proof is mostly determined by the number of branches in the thesaurus. Let $PL(F)$ be the proof length for case frame F containing N case particles. $PL(F)$ adds up all the entropy values of every branch existing between the target noun and the counter category over all examples covered by F.

$$PL(F) = \sum_{Examples} \sum_{Cases} \sum_{Nodes} log_2(Branch) + log_2 N \text{ (bits)}$$

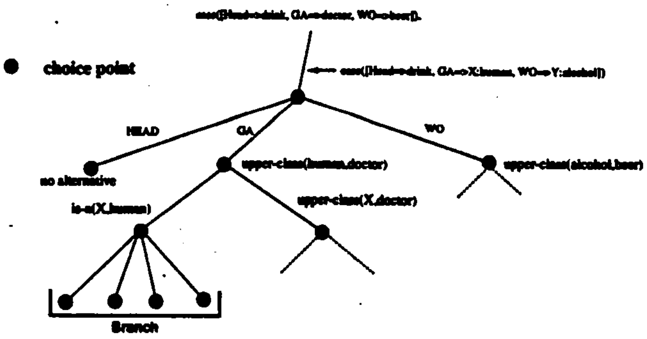
case([Head=>drink, GA=>doctor, WO=>beer]).
choice point
case([Head=>drink, GA=>X:human, WO=>Y:alcohol])
HEAD
GA
WO
no alternative
upper-class(human,doctor)
upper-class(alcohol,beer)
is-a(X,human)
upper-class(X,doctor)
Branch

Figure 2: Choice Points in Proof

Figure 2 illustrates the simple proof strategy adopted by showing how example (4) is proved in terms of case frame (3). After 1 and 2 are given, what we have to do is to prove upper-class(HUMAN,*doctor*) and upper-class(ALCOHOL,*beer*). We trace down the thesaurus from the category of the case frame to the noun of the example. Thus, the number of choice points in the proof amounts to the number of branches of the thesaurus.

If some variable remains free (non bound) and it has n possible categories to take, the proof length for the variable is $log_2 n$, which works as penalty for uselessly long case frames. $PL(F)$ has the following characteristics.

- when F uses general categories, $PL(F)$ tends to be larger because the number of choice points increases.
- when F contains many case particles, $PL(F)$ tends to be larger because the number of choice points increases.

Case Frame Utility

We introduce here a case frame utility measure based on the turing machine model. Although it is not optimal coding, it is well supported empirically. First, we compute the *Explicit-Bits*(F), the encoding length of all examples covered by F. It is the sum of the encoding length $CL(E)$ over each example E. According to Occam's razor principle, a case frame is more plausible when it compresses examples more strongly. We evaluate *utility*(F) of case frame F as the difference between *Explicit-Bits*(F) and the sum of $CL(F)$ and $PL(F)$. *utility*(F) is the net gain in bits when case frame F is introduced. This scheme naturally reflects

[2] This is called Kolmogorov complexity.

the trade-off between case frame generality *Explicit-Bits(F)* and compactness $(CL(F) + PL(F))$. The *utility(F)* is maximized when *Explicit-Bits(F)* and $(CL(F) + PL(F))$ are balanced.

Explicit-Bits(F) $= \sum CL(E)$ (bits) for all E covered by F

utility(F) = *Explicit-Bits(F)* $- (CL(F) + PL(F))$ (bits)

Overall Algorithm

We are now ready to introduce the overall learning algorithm which is briefly depicted in Figure 3. The algorithm is based on a bottom-up covering approach which avoids the combinatorial explosion seen in the exhaustive search method. Let *Ex* be the examples in a list of particle-noun pairs and that *start-point* be a pointer assigned to the beginning of the current example list. First, examples are sorted according to their length. This is based on the intuition that the generalization of longer examples tends to produce useful case frame candidates. Then, s[3] examples (*Sample*) of E starting at *start-point* are selected and their generalizations are produced as a sequence of case frame candidates (*Lggs*). Only s samples at a time are considered to reduce the search space. *Lggs* are evaluated by the compression-based utility measure and the most plausible candidate is stored as a case frame which corresponds to one meaning of a single verb. Next, examples covered by the case frame are deleted from the original list. Case frames are collected by iterating the procedure until the utility falls under a predefined threshold[4]. Once the utility falls under the threshold, the *start-point* is slid towards the shorter examples. This is iterated until *start-point* reaches the end of examples E, producing a set of case frames, each corresponding to one meaning of a verb.

```
Let $s$ be the sample limit
Let Cases be nil
sort-length($Ex, E$)
set start-point at the beginning of $E$
do
  do
    Let Sample($s$) be s samples of $E$ beginning at start-point
    $Lggs = \{C : e, e' \subset Sample(s), C = lgg(e, e')\}$
    $Case = select(Lggs)$
    $Cases = Cases \cap Case$
    $E = E - cover(Case)$
  while ($utility(Case) > threshold$)
start-point = start-point + $s$
while(start-point < |E|)
return   Cases
```

Figure 3: Overall Algorithm

Experimental Results

Preliminary experiments were performed to extract case frames from one year's worth of newspaper articles containing about 75 million words. Table 2 shows the results for 10 typical Japanese polysemous verbs[5]. **No of Examples** and **No of Case Frames** are the number of training examples and the number of case frames acquired by our method, respectively. For evaluating the coverage of the results, a comparison against the hand crafted IPAL dictionary (IPA Lexicon of the Japanese Language for computers) [IPA, 1987] was performed. **IPAL Dic** is the number of IPAL items appearing in the corpus and **RECALL** represents how many of them we acquired from the corpus. Although the result is fairly good, case frames for extremely polysemous verbs such as *kakeru* and *utsu* are not sufficient. This is mainly due to the distribution of examples. In newspaper articles, some fixed expressions appear so often that our algorithm regards other usages that are also general as exceptions. For example, although *kakeru* has a general meaning equivalent to *"hang (something on somewhere)"*, there are only three corresponding examples in 4455 sentences.

No of Case Frames is generally larger than **IPAL Dic**. The reason is the two semantic hierarchies used differ in their grain. All IPAL case frames are described in very general categories (i.e.,ABSTRACT ORGANIZATION), while the frames acquired use all nodes of a 12 level deep thesaurus. In other words, acquired case frames are described in more detailed terms than is true for IPAL. Some idiomatic verb usages that do not have their own entries in the IPAL dictionary[6] were also automatically acquired. Thus, acquired case frames should be useful to disambiguate complex sentences, while the simple case frames like IPAL often fall victim to the ambiguity of real world texts.

Table 2 and Table 3[7] shows the superior case frames[8] of three verbs *noru* and *nomu*. We can conclude the following from these results.

- only essential cases particles are included in the results although we do not incorporate any special mechanism.
- The results well reflects the *co-occurency* existing in the corpus although we do not explicitly handle the

[3] s is currently set at 40.

[4] The threshold is currently set at 100 (bits).

[5] English glosses are attached for convenience. they do not have one-to-one relations with the Japanese verbs.

[6] IPAL dictionary includes idiomatic usages in literal frames, which that cannot discriminate the difference of meaning.

[7] The fifth ranked frame in Table 3 arises because the thesaurus used here describes *water* in a entirely different way. Compilation was aimed at Machine Translation.

[8] The actual Japanese have been replaced by English glosses.

Verb		No of Examples	No of Case Frames	IPAL Dic	RECALL
kakeru (かける)	(*hang*)	4455	32	15	46.7 %
noru (乗る)	(*ride*)	1582	17	6	83.3 %
tatsu (立つ)	(*stand*)	1523	21	6	83.3 %
nozomu (臨む)	(*face*)	650	9	3	100 %
kaku (書く)	(*write*)	604	5	2	100 %
nomu (飲む)	(*drink*)	459	6	4	75.0 %
utsu (打つ)	(*hit*)	448	14	10	50.0 %
tunoru (募る)	(*collect*)	400	7	none	N.A
koeru (越える)	(*exceed*)	283	3	3	100 %
tomonau (伴う)	(*accompany*)	184	4	4	100 %

Table 1: Comparing Acquired Case Frames with IPAL Dictionary

Rank	Utility (bits)	Case Frame
1	8561	NI=>X:VEHICLE, GA=>Y:MAN AND WOMAN
2	5442	GA=>X:ABSTRACT, NI=>*track*
3	824	GA=>X:HUMAN, NI=>*consultation*
4	410	NI =>X:CONCRETE
5	306	GA=>X:HUMAN, NI=>*wind*
6	254	DE=>X:ADMINISTRATIVE REGION, NI=>Y:VEHICLE
7	233	NI=>X:VEHICLE, GA=>Y:HUMAN
8	194	NI=>X:SPIRIT
9	2662	GA=>X:HUMAN, NI=>Y:SITUATION
10	1590	NI=>*boom*

Table 2: Case Frames of *noru* (*"ride"*)

lexical association.

The first is difficult for existing methods to handle because Japanese sentences contain many additional cases and the essential cases are often missed as zero pronouns. The second conclusion is that case frames that compress the examples strongly also have strong co-occurency.

Consider next the details of the acquired case frames and the function of the compression scheme using Table 2 and Figure 4. The first, the fourth, the sixth, and the seventh case frames express the usual meaning of the verb, equivalent to the English verb *"ride"*. The reminders are more or less fixed expressions that frequently appear in newspapers. For example, the second and the third mean *"go well"* and *"advise"*, respectively. Thus, specialized case frames as well as general ones are acquired due to use of compression. The seventh frame is more general than the first frame because category MAN AND WOMAN is three nodes below HUMAN. The compression scheme judges the first as more compact than the seventh due to the distribution of the examples; the increase in the proof length overcomes the increase of generality.

How the compression (in bits) changes according to the number of examples is illustrated in Figure 4. case1, case2 and case3 each correspond to the first, second and third frames of Table 2, respectively. This experiment was performed twice by randomly choosing examples out of the original 1582 examples. Figure 4 plots the average of both trials and illustrates that

- the number of examples has no influence on the priority of case frames.
- the compression increases in a linear manner.

That is, the proposed compression scheme well handles for a wide variety of examples.

Related Work

Many studies have been attempted to automatically acquire verbal case frames from parsed corpora. The main research trend is to use a *lexical association* measure such as *mutual information* [Hindle, 1990; Resnik, 1992] and *t-score* [Hindle, 1991]. The advantage of the approach is the direct introduction of co-occurency as a criteria. This has lead to the extraction of large amounts of lexical knowledge from large corpora. However, the method has the two limitations. One is the difficulty of learning complex structures like multi dimensional case frames. The other is the need for huge amounts of examples. On the other hand, the proposed compression model does not explicitly handle lexical association. It not only handles complex structures in a natural manner, but also learns from a limited data. At the same time, the experimental results guarantee that highly compressive case frames still have strong association.

Rank	Utility (bits)	Case Frame
1	3580	GA=>X:HUMAN, WO=>Y:DRINK
2	818	GA=>X:HUMAN, WO=>Y:MEDICINE
3	267	GA=>X:HUMAN, WO=>Y:LIQUID
4	108	TO=>X:FRIEND AND COLLEAGUE, WO=>*sake*
5	306	GA=>X:HUMAN, WO=>*water*
6	147	GA=>X:ORGANIZATION, WO=>Y:DESIRE

Table 3: Case Frames of *nomu* ("*drink*")

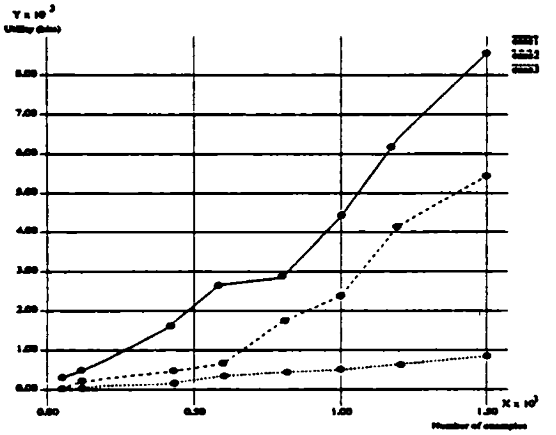

Figure 4: The Compression against the Number of Examples

The proposed compression model provides us with a uniform perspective for rule-based NLP and example-based NLP [Sato and Nagao, 1990] because it discriminates the examples to be encoded in rules from those regarded as exceptions. The serious problem involved in example-based NLP is how the most plausible example is selected, especially when the number of the examples is large. The proposed method will be useful in transforming examples into a more efficient structure by generalizing them.

Conclusion

We have described a new case frame learning method for Japanese polysemous verbs. The two features of the methods are the bottom-up covering algorithm and a newly introduced compression-based utility measure. Experimental results show that the method created a set of well-grained case frames that would require excessive labor if constructed by hand. Although we have focused on verbal case frame acquisition so far, the compression-based approach is promising for other applications.

Acknowledgement

I would like to thank Francis Bond for his helpful discussions.

References

Hassan Ait-kaci and Roger Nasr. Login: A logic programming language with built-in inheritance. *The Journal of logic programming*, 3:185–215, 1986.

Ralph Grishman and John Stering. Acquisition of selectional patterns. In *Proc. 14th COLING*, pages 658–664, 1992.

Donald Hindle. Noun classification from predicate-argument structures. In *Proc. 28th ACL*, pages 268–275, 1990.

Donald Hindle. Structual ambiguity and lexical relations. In *Proc. 29th ACL*, pages 229–236, 1991.

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

IPA. *IPA Lexicon of the Japanese Language for computers IPAL (Basic Verbs) (in Japanse)*. Information-technology Promotion Agency, Japan, 1987.

A. Kolmogorov. Three approaches to the quantitative definition of information. *Problems Inform. Transmission.*, 1:1–7, 1965.

R. Michalski. A theory and methodology of inductive learning. In *Machine Learning: An Artificial Intelligence Approach*, pages 83–134, 1983.

Stephen Muggleton, Ashwin Srinivasan, and Michael Bain. Compression, significance and accuracy. In *Proc. 9th International Conference on Machine Learning*, pages 338-347, 1992.

P. Resnik. Wordnet and distributional analysis. In *Proc. AAAI Workshop on Statistical Methods in NLP*, pages 268–275, 1992.

Satoshi Sato and Makoto Nagao. Toward memory-based translation. In *Proc. 13th COLING*, pages 247–252, 1990.