A Left-to-right Breadth-first Algorithm for Subcategorization Frame Selection of Japanese Verbs

Kazunori Muraki, Shin'ichiro Kamei, Shinichi Doi

NEC Corporation
Information Technology Research Laboratories
4-1-1, Miyazaki, Miyamae-ku, Kawasaki 216, JAPAN
e-mail: {k-muraki, kamei, doi}@hum,cl.nec.co.jp

Abstract. Ambiguity resolution for verbs with their subcategorization frames is crucial for natural language processing systems. Japanese is an agglutinative language and the case relations between predicates and their arguments are marked by case marking postpositions. This agglutinativeness makes Japanese a free word order language. In addition, although the case marking postpositions are the crucial keys to select subcategorization frame and decide deep cases in Japanese, they can easily overwritten, changed or omitted through the linguistic operations such as passivization, topicalization, relativization. We propose a breadth-first algorithm to enable efficient matching and the disambiguation of the subcategorization frames. The representation for the subcategorization frame is a combinatorial encoding of surface case frame and the deep case structure. Its design aims at eliminating possible redundancies due to the above mentioned characteristics. The proposed mechanism and the lexical representation has been used and evaluated in a couple of commercially available MT applications that translate Japanese to English.

1 Introduction

Ambiguity resolution for verbs with their subcategorization frames is crucial for an MT system to produce readable translations. It is because the failure in the disambiguation will lead to generating a translation for a completely different sentence in many cases. Thus, highest priorities in MT systems development often lie in accuracy of the subcategorization frame descriptions in the lexicon and the matching algorithm that takes advantages of lexical and contextual valences such as selectional restrictions and other richer lexical information [Boguraev & Pustejovsky, 1992].

We have adopted a 'quality first, quantity next' approach to develop the lexicon and the heuristic matching rules as is opposed to knowledge mining approach of extracting subcategorization frames out of MRDs and corpora [Manning, 1991]. The 'quality first approach' starts with analysis and classification over multiple subcategorization frames, their derivative forms and their mutual relationships often with more than one word senses. Our encoding system for Japanese verbs is characterized by designing the codes for the combination of surface case frame and the deep case frame (alias, thematic role frame or the argument structure), both in canonical forms. With some ideas such as alternative surface/deep case markers for a slot in the frame [Nomura & Muraki, 1988], the number of the different subcategorization frame code has converged at 250 through encoding 30,000 verbs. The encoding has been done by a bootstrapping method that involves human lexicographers empowered by the set of current subcategorization frame codes and the already encoded examples instantly shown by a computer program utility.

Besides the efforts to create less redundant, more manageable and more powerful lexicon, we have designed and improved a breadth-first deterministic procedure for verb subcategorization frames to be matched and disambiguated in context. The procedure is of an original design for

a free word ordering language - Japanese, and is designed to be free of excessive complexities added by extra mechanisms to parsers for configurational languages [Karttunen & Kay, 1985]. The following section examines the nature of Japanese verb subcategorization frames and sees how the elliptical qualities of case markers turn the frames into the subject of disambiguation rather than the source information useful in the disambiguation. The main section (Section 3) describes the procedure for the parser to generate the subcategorization frames for higher predicates, to match and evaluate the plausibility of the frames in context, and to perform the final heuristic selection of the most plausible frame using pragmatic conditions.

2 Subcategorization Frame Ambiguities in Japanese and the Lexicon Encoding Framework

Japanese is an agglutinative language and the case relations between predicates (verbs, adjectives, etc.) and their arguments (mainly noun phrases) are marked by case marking postpositions, such as "-ga", "-wo" and "-ni", which often mark nominative, accusative, and dative cases, respectively. The representation for the subcategorization frame is a combinatorial encoding of surface case frame and the deep case structure. For example, the Japanese verb "ageru" meaning 'give' has a subcategorization frame like [ga - AGenT][wo - PATient][ni - GOA1], which means that "ageru" takes 3 arguments and the deep case or thematic role of the noun phrase with postposition "-ga" is agent, the one of NP with "-wo" is patient, and the one of NP with "-ni" is goal. Therefore, the Japanese sentence "X-ga Y-wo Z-ni ageru" is translated to 'X gives Y to Z.'

This agglutinativeness provides Japanese with one of its most notable syntactic characteristics so called scrambling, the phenomenon in which word ordering is almost free for the syntactic elements in a Japanese simple sentence except for the predicate phrase that is placed at the end of the sentence. All the examples in e.g. 2-1 lead to the same event structure interpretation and are translated to 'X gives Y to Z.'

```
e.g. 2-1 a. "X-ga Y-wo Z-ni age-ru" b. "Y-wo X-ga Z-ni age-ru"
          c. "Y-wo Z-ni X-ga age-ru" d. "X-ga Z-ni Y-wo age-ru"
          e. "Z-ni X-ga Y-wo age-ru" f. "Z-ni Y-wo X-ga age-ru"
```

In addition, Japanese has another syntactic characteristics that concern the qualitative and quantitative nature of the verb subcategorization frames, the totally elliptical quality of these case elements. Any adverbial NPs in a simple Japanese sentence can appear at any position or do not have to appear at all even when they are to be assigned nominative or accusative cases. So, any combination of elliptical "X-ga", "Y-wo" and "Z-ni" in any of the sentences e.g. 2-la. through e.g. 2-1f is grammatical as well. Traditionally, these characteristics have been considered to be a major engineering burden for parsers.

The identity of the subcategorization frame is endorsed by the identical mapping pattern between the surface cases, namely the case postpositions, and the deep cases among all those syntactic variations. Since any "X," in e.g. 2-1 is mapped to a deep case AGenT, "Y" to a PATient and "Z" to a GOA1, we have only one subcategorization frame code VT31 (Verb with 3 arguments, Typel) in the lexicon for all the six examples in e.g. 2-1. The code VT31 defines the canonical form of the frame with the standard word ordering shown in e.g. 2-la. It also has alternative surface/deep case markings for some slots: e.g. the DATive case slot is [ni/e - GOAl], which allow an alternative surface case marker "he" to fill in and occupy the slot. Eventually, VT31 covers 6x2+(2+4+4)+(l+l+2)+l = 27 surface case occurrence patterns.

Thus, our subcategorization frame sets can reduce the difficulty of frame selection caused by scrambling and ellipsis.

Other difficulties in subcategorization frame selection of Japanese verbs are the multiple subcategorization frames for a single Japanese verb and the elliptical or changeable qualities of case markers. Although the case marking postpositions are the crucial keys to select subcategorization frame and decide deep cases in Japanese as we described, they can easily overwritten, changed or omitted through the linguistic operations such as passivization, topicalization, relativization. An efficient algorithm for matching and disambiguation of the subcategorization frames are required to overcome these difficulties.

Multiple subcategorization frames for a single Japanese verb typically occurs when there are more than one word senses or when some deeper generative lexical semantic mechanisms seem to be functioning within the lexicon as is observed in e.g. 2-2 [Levin, 1993], [Nomura et al., 1994].

```
e.g. 2-2  a.      "Taro-ga hana-wo mado-ni kazatta."
           Taro-NOM flower-ACC window-DAT decorated-PERF.
          'Taro arranged the flowers on the window (as the decoration).'
          b.    "Taro-ga mado-wo hana-de kazatta."
              Taro-NOM window-ACC flower-WITH decorated-PERF.
              'Taro decorated the window with the flower.'
```

Since the ACCusative cases in both e.g. 2-2a and e.g. 2-2b is mapped to a deep case PATient, these subcategorization frames are not only different from each other, but also are incompatible with each other. In effect, there have to be at least two word senses for the verb "kaza-ru" in order not to intermingle the two independent case elements 'window' and 'flower.'

Even more frequently occurs a predicate phrase with multiple subcategorization frames when it is used with voice auxiliary verbs and/or equivalents. For the purpose of parsing efficiency, our system models Japanese as having extended category of voice conversion with more than a dozen conversion pattern codes in addition to ordinary ones such as passivization. The idea is to reduce the structures of higher predicates such as causative construction into a flat construction with its surface case markings permuted. The variety of the extended voice auxiliary verb category roughly corresponds to the variety of English auxiliary verbs and frequent higher predicates 'let' 'make,' and 'want.' A Japanese auxiliary verb "reru/rareru" that corresponds to an English auxiliary verb 'can' or 'be +pp.' (passivization) acts exactly the same even when it means 'possibility.'

```
e.g.2-3  a.  "X-ga  Y-wo tabe-ru." b. "X-ni  Y-ga  tabe-rareru."
              X-NOM Y-ACC eat         X-DAT Y-NOM    eat-RARERU
             'X eats Y'              'X can eat Y'  xor 'X is eaten by Y'
```

As is observed in e.g. 2-3, the nominative case marker "-ga" turns into dative case marker "-ni," when the passive/potential auxiliary verb "reru/rareru" is attached. Likewise, the accusative case marker "-wo" turns into nominative case marker "-ga." Our collection of syntactic facts of this kind suggested that these phenomena should be uniformly and efficiently treated as permutations of the surface case set in the verb subcategorization frame.

3 The Procedure

The basic parsing strategy is, first to generate multiple frames by permuting case markers recursively, and then to match the frames in context with a minimal number of parsing head

movement. Matching is to be done deterministically through one slot filling at a time by evaluating all the remaining subcategorization frames at a time. The final stage of the frames ambiguity resolution consists of heuristic sub-procedures using fragments of pragmatic information in context. The following subsections describe the three processing stages in sequence.

3.1 Generation of Subcategorization Frames

When the morphological analyzer detects the existence of voice auxiliary verbs or equivalents while checking the predicate phrase, the analyzer develops the subcategorization frame code such as VT31 into the form of surface case - deep case mapping frame. SCPF (Surface Case Permutation Frame) codes for the voice auxiliary verbs are loaded in the memory as well. When all the necessary information is ready, the analyzer generates the subcategorization frames for the predicate. The process consists of one permutation for one auxiliary verb at a time. The first permutation is performed for the first auxiliary verb next to the main verb, and the focus moves on from the main verb to the first auxiliary verb. The N-th permutation is performed for the N-th auxiliary verb next to the (N-l)-th auxiliary verb, and the focus moves on from the (N-l)-th auxiliary verb to the N-th auxiliary verb. The maximum number for N is set to three in our MT system, considering the practical complexities of case permuting higher predicates in real utterances and written sentences.

```
e.g.3-3  a. "X-ga Y-wo taberu."
             X-NOM    Y-ACC eat
            'X eats Y'
         b. "Z-ga Y-wo X-ni tabe-saseru."
             Z-NOM Y-ACC Z-DAT eat-CAUS
            'Z makes X to eat Y'
         c. "X-ga Y-wo Z-ni tabe-sase-rareru."
             X-NOM    Y-ACC    Z-DAT    eat-CAUS-PASS
            'X was made to eat Y by Z'
```

The correct process should generate the subcategorization frames represented in the example sentences from e.g. 3-3a through e.g. 3-3c, where all case elements X, Y and Z are consistent in these three. The SCPF codes are developed into surface case permutation frame: examples for the causative auxiliary verb “saseru” are shown in Fig.l.

	permutation commands			permutation commands	
Causative A	NULL=: NOM(CAUser) NOM =: DAT		Causative B	NULL=: NOM(CAUser) NOM =: ACC	

Fig.l SCPF Codes and Permutation Commands for Auxiliary Verb "saseru"

Using a small linguistic knowledge table that filters the possible sequential combination of a subcategorization frame code (e.g. VT31) and SCPF codes, the SCPF code ‘Causative A’ is selected and the two permutation commands are executed. The command ‘NULL=:

NOM(CAUser)' adds to the frame a new deep case 'CAUser' and marks it with NOMinative case. The command 'NOM =: DAT' turns "X-ga" into "X-ni." Thus, the original frame for e.g.3-3a turns into the frame for e.g.3-3b as the result of the first case permutation. The second permutation to derive e.g.3-3c takes place the same way as the first one, except that it uses another filter between a SCPF code and the SCPF codes that can follow the first code.

The general algorithm for the generation of subcategorization frame is described in the following procedure.Generation. It covers the occurrences of any number of multiple frames at any point in the generation, including the original multiple frames in the lexicon.

```
procedure.Generation
  while  the set {Subcat Frame Code} is not empty
        'take one Subcat Frame Code and develop it into the frame';
        'fill in each slot the selectional restrictions from the dictionary';
        if  'there is an auxiliary verb with SPCF codes next to
             the current focus verb or auxiliary verb'
        then {'duplicate the frame by the number of SPCF codes';
              'permute the surface case by each permutation command'   ;}
        else 'subtract the Subcat Frame Code from {Subcat Frame Code}'
    end
end
```

3.2 Breadth-First Algorithm for Matching Valences of the Multiple Subcategorization Frames and the Words in Context

Our push-down, shift-reduce parser is context-sensitive with four windows. The Main Focus Window '*' is next to the right most window '+' called Right Context Window. The two others '-' and '=' are called Left Context Window and Farther Left Context Window, respectively. The reason for doubling the number of left context windows is simply that Japanese is the verb final, left-branching language.

The following procedure, Matching describes the overall procedure for the parser to match the frames in context with a minimal number of parsing head movement. It is to be done deterministically through one slot filling at a time by evaluating all the remaining subcategorization frames,

```
'=' : Farther Left Context Window
'-' : Left Context Window
'*' : Main Focus Window
'+' : Right Context Window

procedure.Matching
  while {Word List} not empty
        'Move the Main Focus '*' on to the main verb candidate while reducing
         the local constituents within NPs';
        while { Subcat Frame } is not empty and either '=' or '-' is not NULL
          'take one Subcat Frame with maximal number of slots';
           while the Subcat Frame is effective and {all the slots in the
              Subcat Frame are not filled or either '=' or '-' is not NULL}
           'Check if the case element at '-' position matches with one of the
             surface cases and the selectional restrictions in a slot of the
```

```
             Subcat Frame ;
            if    Matching is successful
            then   'fill the slot'  ;
            else   'abandon the Subcat frame';
          end
          Call procedure.EvaluateSubcatFrame;
          'reduce '-' and '*' into new '*'   ';   * '=' becomes new '-'
        end
        if    { Subcat Frame } is empty
        then  'resurrect once maximally filled Subcat frames
               out of the abandoned frames';
        else
        if    { Subcat Frame } has more than one element
           then
               { Call procedure.EvaluateSubcatFrame;
                 Select the most plausible Subcat Frame;  }
        'reduce  '-'  and '*' into new '*'   '; #  '=' becomes new '-'
end

procedure.EvaluateSubcatFrame
begin
if  '=' contains obligatory case (NOM | ACC) markers
  then
       {'find Subcat frames without the obligatory case marker in '='  ';
       'reduce the plausibility score of the found frames by 95%  '; }
'add weighted points to the filled slots  '; # as ACC and NOM weighs higher
'subtract weighted points from the unoccupied slots ';
end
```

The point in procedure.Matching is to delay the shift-reduce operation until all the matching and plausibility evaluations on the remaining subcategorization frames are completed. This is why we call it a breadth-first algorithm. The focus window of the parser does not have to move around to find the best match between the set of multiple subcategorization frames and the set of case elements in context. Especially when the system parses frequent sentence constructions where higher predicate structures could be reduced into the extended category of voice auxiliary verbs, the parser could perform the matching with a lot less computational cost than naively applied CFG parsers.

3.3 Heuristic Disambiguation of the Verb Phrases in Context

The most frequent case in which multiple subcategorization frames still remain after the matching procedure described in the previous section is the case with one of the most popular Japanese auxiliary verbs, “reru/rareru”. As is examined in section 2, “reru/rareru” has the ambiguities of at least passive and possibility, which share the several subcategorization frames with each other.

```
e.g.3-3  a.  "kono ringo-wa (ga) watasi-ni-wa tabe-rareru."
              this  apple-NOM     me-DAT       eat-RARERU
             'This apple is edible for me.'
```

```
b.  "kono ringo-wa (ga) kare-ni tabe-rare-ta."
    this  apple-NOM     him-DAT eat-RARERU-PERF
   'This apple was eaten by him.'
```

As are observed in e.g.3-3, the differences in focus, topic relations and aspect seem to affect the disambiguation results in addition to the traditional usage of selectional restrictions and other lexical semantic features. Instead of marking cases, special arrangements of word ordering within Japanese predicate phrase seem to be useful in extracting pragmatic effects mentioned above. The use of topic postpositions such as “wa” seems to compensate the lack of articles and, hence, the distinctions of definiteness/indefiniteness so that it could be used as another clue of discourse to disambiguate among the remaining subcategorization frames such as e.g.3-3 a and b.

Through an empirical study and the evaluations on the heuristic conditions for the final stage of subcategorization frames disambiguation, we have developed disambiguation procedures for ambiguous auxiliary verbs and equivalents. These procedures take advantage of visible clues described above. Below is the most frequently triggered lexical heuristic procedure for auxiliary verb “reru/rareru.”

```
Procedure.Heuristics.reru
if         Accusative case exists in the scope
then
          if DATive case precedes Accusative case
            then Choose 'Passive' and the corresponding Subcat frame;
        else
           if Predicate has PERFective tense/aspect
           then Choose 'Passive' and its Subcat frame;
           else Choose 'Possibility' and its Subcat frame;
else
        if NOMinative case precedes DATive case
        then
           if Predicate has PERFective tense/aspect
           then Choose 'Passive' and its Subcat frame;
           else
                    if Predicate is in nominalized phrase
                 then
                      if the nominalized phrase constructs cleft sentence
                          then Choose 'Possibility' and its Subcat frame;
                          else Choose 'Passive' and its Subcat frame;
                  else Choose 'Passive' and its Subcat frame;
         else
           if topic postposition "-wa" is attached to DATive case
           then Choose 'Possibility' and its Subcat frame;
           else
                    if Predicate is in nominalized phrase
                 then Choose 'Possibility' and its Subcat frame;
                 else
                            if Predicate has PERFective tense/aspect
                       then Choose 'Passive' and its Subcat frame;
                          else Choose 'Possibility' and its Subcat frame;
end
```

Procedure.Heuristics.reru can provide only the default disambiguation results so that more global contextual information and/or sophisticated lexical semantic models [Pustejovsky, 1991] should be added to the conditions in the procedure. The current score of the accuracy in average is 74%. A good quality of the proposed procedural representation of the heuristic knowledge is that the conditions are well localized so that one can trace the disambiguation process easily and improve the accuracy.

4 Conclusion

We have proposed a lexical representation model and a breadth-first procedure to efficiently disambiguate Japanese predicate phrases with multiple subcategorization frames. The procedure coupled with the combinatorial subcategorization frame representation takes advantage of the qualities such as free word ordering and elliptical cases for improving the matching efficiency rather than adding extra mechanisms to parsers for configurational languages. This lets the parser have an equivalent amount of information processing performed by a CFG parser, especially when the higher predicate structures are reduced.

Not only the general procedure for matching and plausibility evaluation of the subcategorization frames, but also heuristic procedures using pragmatic information for the final selection have been developed for each ambiguous auxiliary verb. The Japanese parser turned out to be not only feasible but also practical as could be used in a couple of commercially available MT applications that translate Japanese to English with more than 100,000 vocabulary each.

Future works include making the matching algorithms more flexible as in the sense originally described in [Pustejovsky, 1991] as 'type coercion'. Improvement in accuracy would require taking more advantage of statistic factors into the open heuristic procedures for one thing. Another is the use of contexts in the form of processing history for the subcategorization frame matching procedure.

References

[Boguraev & Pustejovsky, 1992] Boguraev, Branimir and James Pustejovsky. 1992. Lexical Ambiguity and The Role of Knowledge Representation in Lexicon Design, In *30th Annual Conference of the Association for Computational Linguistics,* Newark, Del., pp.36-41.

[Dorr, 1993] Dorr, Bonnie J. 1993. *Machine Translation - a View from the Lexicon,* MIT Press.

[Karttunen & Kay, 1985] Karttunen, Lauri and Martin Kay. 1985. Parsing in a Free Word Order Language, In *Natural Language Parsing* edited by Dowty, D. et al, Cambridge University Press.

[Levin, 1993] Levin, Beth. 1993. *English Verb Classes and Alternations A Preliminary Investigation,* The University of Chicago Press.

[Manning, 1991] Manning, Christopher. 1991. Automatic Acquisition of a Large Subcategorization Dictionary from Corpora, In *29th Annual Conference of the Association for Computational Linguistics,* Berkeley, Calif., pp.235-242.

[Muraki, 1987] Muraki, Kazunori. 1987. PIVOT: A Two Phase Machine Translation System, In *MT Machine Translation Summit: Manuscripts and program,* Hakone, Japan, pp.81-83.

[Nomura & Muraki, 1988] Nomura, Naoyuki and Kazunori Muraki. 1988. Case Frame Model of Machine Translation System PIVOT, In *Proceedings of 38th IPSJ Conference,* Tokyo, Japan, pp.390-391. (in Japanese)

[Nomura et al., 1994] Nomura, Naoyuki, Doug Jones, and Robert C. Berwick. 1994. An Architecture for a Universal Lexicon, In *COLING94: The 15th International Conference on Computational Linguistics,* Kyoto, Japan, pp.243-249.

[Pustejovsky, 1991] Pustejovsky, James. 1991. The Generative Lexicon. *Computational Linguistics,* 17. pp.409-441.