Acquiring Reliable Predicate-argument Structures from Raw Corpora for Case Frame Compilation

Daisuke Kawahara[†], **Sadao Kurohashi**[†‡]

[†]National Institute of Information and Communications Technology
3-5 Hikaridai Seika-cho, Soraku-gun, Kyoto, 619-0289, Japan
[‡]Graduate School of Informatics, Kyoto University
Yoshida-Honmachi, Sakyo-ku, Kyoto, 606-8501, Japan
dk@nict.go.jp, kuro@i.kyoto-u.ac.jp

Abstract

We present a method for acquiring reliable predicate-argument structures from raw corpora for automatic compilation of case frames. Such lexicon compilation requires highly reliable predicate-argument structures to practically contribute to Natural Language Processing (NLP) applications, such as paraphrasing, text entailment, and machine translation. We first apply chunking to raw corpora and then extract reliable chunks to ensure that high-quality predicate-argument structures are obtained from the chunks. Our experiments confirmed that we succeeded in acquiring highly reliable predicate-argument structures on a large scale.

1. Introduction

Predicate-argument structures (also known as logical forms and case structures) represent what arguments are related to a predicate, and form basic units for conveying the meaning of natural language text. Identification of such predicate-argument structures plays an important role in natural language understanding.

To precisely identify predicate-argument structures, selectional preferences are necessary. Selectional preferences are a kind of linguistic knowledge that what arguments can have a relation to each predicate. One of such knowledge sources is *case frames*. Thus far, semantic case frames, such as FrameNet (Baker et al., 1998) and PropBank (Palmer et al., 2005), in which each frame is semantically disambiguated, have been elaborated manually. However, they do not provide sufficient selectional preferences since there are a few descriptions of words or semantic markers that can fill each case slot.

On the other hand, knowledge acquisition from large corpora has attracted attention in recent years. In particular, many approaches to automatically acquire case frames have been proposed. However, most of these approaches focused on subcategorization frames (e.g., (Brent, 1993; Manning, 1993; Briscoe and Carroll, 1997; Korhonen and Preiss, 2003)), which are syntactic case frames representing argument patterns of verbs. Therefore, these subcategorization frames neither distinguish verb senses nor provide selectional preferences[1].

This study aims at automatically compiling semantic case frames for English predicates, such as the FrameNet (Baker et al., 1998), from a large raw corpus. For example, let us show a case frame of the verb "arrest":

arrest
sbj:{police, authority, ...} *obj*:{people, suspect, ...}
pp:on:{charge, suspicion, ...}

Frequencies are attached to each case frame, case slot, and word. These frequencies can be effectively utilized in various applications of case frames, such as parsing, paraphrasing, and machine translation.

In this study, we adopt the following strategy that was used in (Kawahara and Kurohashi, 2006; Kawahara and Uchimoto, 2008): first extract reliable predicate-argument structures from large raw corpora, and then compile semantic case frames from these predicate-argument structures. The most important issue to be addressed here is how to extract as reliable predicate-argument structures as possible to yield high-quality case frames. As stated above, however, to precisely identify predicate-argument structures, case frames are required. This means a chicken and egg question. In this paper, we propose the initial step of extracting reliable predicate-argument structures without case frames.

2. Related work

Subcategorization frames are closely related to our case frames. Subcategorization frames are a class of case frames and represent generalized argument patterns of verbs. For example, a subcategorization frame for the verb 'put' is "NP put NP PP," which implies that 'put' takes a noun phrase (NP) as its subject, and an NP and a prepositional phrase (PP) as its complements. Subcategorization frames were constructed manually in the early stages of NLP (Boguraev et al., 1987; Grishman et al., 1994; The XTAG Research Group, 1998). These manually constructed lexicons were used as a gold standard when evaluating automatic construction approaches, which are stated below.

The first methods that automatically learn subcategorization frames from corpora were proposed by Brent (Brent, 1993). These methods focused on a small number of predefined subcategorization frames. Subsequent approaches

[1] Originally, subcategorization frames do not provide selectional preferences, but it is possible to preserve words that constitute these frames, as shown in (Korhonen et al., 2006). These words can be used as selectional preferences.

targeted larger sets of predefined subcategorization frames and used a larger amount of corpora (Ushioda et al., 1993; Manning, 1993; Ersan and Charniak, 1996; Gahl, 1998; Carroll and Rooth, 1998; Lapata, 1999). Another challenging system automatically detected a set of subcategorization frames and constructed a lexicon of them (Briscoe and Carroll, 1997). To extract relevant subcategorization frames for each verb, many of the previous approaches made use of hypothesis testing. However, it was reported to have poor performance, especially for low-frequency subcategorization frames (Briscoe and Carroll, 1997; Manning and Schütze, 1999). Furthermore, verb sense ambiguity, which was not distinguished by these systems, was a cause of the poor performance. Recently, Korhonen et al. proposed a sophisticated method that integrates improved hypothesis testing and word sense disambiguation (Korhonen, 2002; Korhonen and Preiss, 2003).

On the other hand, semantic case frames have been manually elaborated in the projects such as FrameNet (Baker et al., 1998) and PropBank (Palmer et al., 2005). Kawahara and Kurohashi succeeded in automatically constructing Japanese semantic case frames from a large Web corpus (Kawahara and Kurohashi, 2006). They first applied Japanese-specific rules to extract reliable predicate-argument structures from automatic parses. Then, they clustered the predicate-argument structures to produce case frames on the basis of a thesaurus. This study makes use of Japanese characteristics for compiling precise Japanese case frames, such as the head-final nature and explicit case-marking postpositions.

Kawahara and Uchimoto proposed a method for automatically compiling case frames for English (Kawahara and Uchimoto, 2008). They first applied dependency parsing to an English corpus, extracted predicate-argument structures and applied clustering to them on the basis of WordNet. To extract reliable predicate-argument structures, they simply used relatively short (20 words or less) sentences. However, this method is too naive to obtain reliable predicate-argument structures as mentioned in section 3.

To acquire high-quality parses from the outputs of parsers, Reichart and Rappoport proposed an ensemble method (Reichart and Rappoport, 2007). They regarded parses as being of high quality if 20 different parsers agreed. Ravi et al. proposed a method for estimating parse accuracy (Ravi et al., 2008). They used an SVM regression approach on the basis of text-based and parse-based features.

3. A method for acquiring reliable predicate-argument structures

We acquire reliable predicate-argument structures from raw corpora. The predicate-argument structure of our target consists of a predicate (abbreviated as "pred" in the following examples) and one or more arguments. The arguments are classified into five classes: "sbj" (subject), "obj" (direct object), "obj2" (indirect object), "sbar" (sentential complement), and "pp" (prepositional phrase). Here is an example of predicate-argument structures to be acquired:

(1) sbj:[I] pred:[borrow] obj:[the kits]
pp:with:[a $ 25.00 deposit]

To acquire such predicate-argument structures, it is necessary to identify a subject noun phrase (NP), a verb phrase (VP) as a predicate, object NPs, and prepositional phrases (PPs). Therefore, it is straightforward to identify these phrases by applying chunking.

In order to acquire as reliable predicate-argument structures as possible from chunking results, we discard unreliable and inappropriate chunks for our purpose. Our method of acquiring reliable predicate-argument structures consists of the following three steps:

1. apply chunking to a raw corpus,
2. filter out unreliable and inappropriate sentences and chunks,
3. extract predicate-argument structures, and apply prepositional phrase attachment disambiguation if a prepositional phrase exists.

In the following subsections, we describe these three steps in detail.

3.1. Chunking

We apply chunking to a large raw corpus. Before chunking, it is necessary to assign part-of-speech tags by tagging (including tokenization). To carry out these processes, we use Tsuruoka's tagger[2](Tsuruoka and Tsujii, 2005) and an SVM-based chunker[3](Kudo and Matsumoto, 2001). We trained this chunker on sections 2-21 of the Penn Treebank (Marcus et al., 1994).

To evaluate the accuracy of this chunker, we applied the chunker to the development set (section 22) of the Penn Treebank that was automatically tagged by the abovementioned tagger. With the chunker, a precision of 93.89% and a recall of 93.06% were achieved. The accuracies of NP, PP[4] and VP, which are most related to the acquisition of predicate-argument structures are listed below.

type	precision	recall	F1
NP	94.23%	94.02%	94.13
PP	96.75%	97.98%	97.36
VP	94.29%	92.50%	93.39

Kawahara and Uchimoto used only short (20 words or less) sentences to obtain reliable parses (predicate-argument structures) (Kawahara and Uchimoto, 2008). However, the accuracies of NP, PP, and VP of short sentences in section 22 decreased as shown below.

type	precision	recall	F1
NP	93.61%	93.36%	93.49
PP	95.87%	96.79%	96.33
VP	92.40%	89.40%	90.88

This result indicates that long sentences are not necessarily prone to make chunking errors.

For example, let us consider the following sentence:

[2] http://www-tsujii.is.s.u-tokyo.ac.jp/~tsuruoka/postagger/

[3] http://chasen.org/~taku/software/YamCha/

[4] The PP that is identified by the chunker is different from the definition of the argument "pp," which consists of a pair of adjoining PP and NP.

(2) I borrowed the kits with a $25.00 deposit.

From this sentence, we obtain the following chunks:

(3) NP:[I] VP:[borrowed] NP:[the kits] PP:[with] NP:[a $ 25.00 deposit]

3.2. Filtering out unreliable and inappropriate chunks on the basis of linguistic characteristics

To acquire reliable predicate-argument structures, we filter out unreliable and inappropriate chunking results on the basis of linguistic characteristics. We boldly discard unreliable and inappropriate chunking results for our purpose, but to guarantee the massive quantity and variation of resulting predicate-argument structures, we use an extensive amount of raw corpora as stated in section 4.

We use linguistic rules of discarding the following unreliable and inappropriate sentences and chunks for the acquisition of predicate-argument structures.

- sentences to be discarded
  - a sentence that begins with a VP or a PP
  - a sentence that ends with a question mark
  - a sentence that has a comma being adjacent to a VP
  - a sentence that contains a sign (e.g., "–", ";")
  - a sentence that does not have an NP before a VP
  - a sentence in which the first VP is a participle or an infinitive
- chunks to be discarded
  - the chunks following the first comma outside an NP
  - the chunks following wh-clauses
  - the chunks following the second VP except participles and infinitives

These heuristics are applied to guarantee that each remaining sentence contains a predicate and at least one argument and to cut off complex parts of sentences that are prone to make errors.

Below, we show the chunking accuracies in section 22 of the Penn Treebank after filtering out sentences and chunks that obey the abovementioned rules.

type	precision	recall	F1
NP	96.18%	95.00%	95.59
PP	97.51%	97.51%	97.51
VP	96.46%	94.34%	95.39

After this filtering, we acquired 2,679 NP, PP, and VP chunks from the development set. Since we acquired 14,975 chunks of these types before filtering, the acquisition rate was 17.9%. This ratio may sound low, but we use a massive amount of raw corpora to compensate for this low coverage.

We performed error analysis on the incorrectly extracted VP and NP chunks. The precision of VPs was 517 / 536 (0.9646), indicating that 19 VPs were incorrectly extracted. Manual investigation of these 19 VPs revealed that 12 of them were not harmful for the acquisition of predicate-argument structures. For example, the VP chunk "successfully contended" was judged to be a combination of two chunks "successfully" (ADVP) and "contended" (VP). However, such a gap does not affect the acquisition of predicate-argument structures since adverbs are ignored to make a predicate representation (described in section 3.3.). If we consider these incorrect chunks to be correct, the precision of VPs becomes 529 / 536 (0.9869).

The remaining errors involve really difficult cases. For example, the following sentence contains a chunking error:

(4) His firm favors selected computer, drug and pollution-control stocks.

From this sentence, the tools that we adopted extracted an incorrect VP "selected", whereas the correct VP is "favors." This sentence and these chunks were not filtered by our current heuristics. Hence, we need to specify patterns to filter out these errors in the future.

The precision of NPs was 1559 / 1621 (0.9618); 62 NPs were incorrectly extracted. Again, manual investigation of these 62 NPs revealed that 38 of them were not harmful for the acquisition of predicate-argument structures. The major errors were caused by the handling of figures. For example, "about 10,000 diamond miners" was automatically identified as an NP, but the gold standard is two NPs ("about 10,000" and "diamond miners"). In such cases, automatic results are more appropriate for the acquisition of predicate-argument structures. If we consider these incorrect chunks to be correct, the precision of NPs becomes 1597 / 1621 (0.9852).

Accordingly, we succeeded in acquiring reliable chunks with an accuracy of around 98%.

3.3. Extracting predicate-argument structures from chunks

We extract predicate-argument structures from the filtered results of automatically detected chunks. We use the following straightforward rules to convert chunks into a predicate-argument structure.

- VP → "pred"
- NP preceding the predicate → "sbj"
- NP following the predicate → "obj"
- NP following the direct object → "obj2"
- SBAR → "sbar"
- a pair of adjoining PP and NP → "pp"

Extracted predicates are lemmatized, and modal verbs and adverbs in the predicates are deleted. For example, let us consider the following sentence again:

(5) I borrowed the kits with a $25.00 deposit.

From this sentence, we obtained the following chunks:

(6) NP:[I] VP:[borrowed] NP:[the kits] PP:[with] NP:[a $ 25.00 deposit]

From these chunks, the following predicate-argument structure is extracted:

Table 1: Examples of acquired predicate-argument structures.

sbj:[the super-user] pred:[raise] obj:[the hard limits]
sbj:[it] pred:[strengthen] obj:[the action]
sbj:[he] pred:[raise] obj:[a hand]
sbj:[this web page] pred:[be linked] pp:to:[any other web sites]
sbj:[a user] pred:[view] obj:[items] pp:from:[your catalog]
sbj:[you] pred:[read] obj:[this]

(7) sbj:[I] pred:[borrow] obj:[the kits]
pp:with:[a $ 25.00 deposit]

For prepositional phrases, we must determine their head (predicate or object). We employed a method of PP attachment disambiguation based on a large number of unambiguous examples extracted from a raw corpus (Kawahara and Kurohashi, 2005). If the head of a prepositional phrase is judged as an object, it is discarded.

4. Experiments

To acquire reliable predicate-argument structures, we used a Web corpus comprising of 2 billion English sentences as a source corpus. This Web corpus was crawled and constructed in the same manner mentioned in (Kawahara and Kurohashi, 2006). We applied the above method to this corpus and acquired 2.4 billion predicate-argument structures. Table 1 shows some examples of the acquired predicate-argument structures.

We evaluated 200 predicate-argument structures that were randomly selected from the acquired results. We obtained an accuracy of 97%. Major errors were caused by incorrect objects of "say" and "know", which were extracted from sentences in which a complementizer ("sbar") was omitted. Automatically detected "sbar" also had errors. That is, appositive markers were incorrectly judged as "sbar." Another type of errors was caused by PP attachment disambiguation.

We plan to improve the acquisition of predicate-argument structures by employing an iterative process, that is the incorporation of the knowledge gained from the resulting case frames into the identification of predicate-argument structures.

5. Conclusion

This paper has described a method for acquiring reliable predicate-argument structures from a large raw English corpus. Experimental results showed that we succeeded in acquiring reliable predicate-argument structures. Thus, we are ready to compile case frames from them. We will complete compiling wide-coverage case frames and make them freely available on the Web. These case frames can be used in many NLP analyzers as well as in NLP applications such as parsing, text entailment, paraphrasing, and machine translation.

6. References

Collin Baker, Charles J. Fillmore, and John Lowe. 1998. The Berkeley FrameNet Project. In *Proceedings of COLING-ACL1998*, pages 86–90.

Bran Boguraev, Ted Briscoe, John Carroll, David Carter, and Claire Grover. 1987. The derivation of a grammatically-indexed lexicon from the Longman Dictionary of Contemporary English. In *Proceedings of the 25th Annual Meeting of the Association for Computational Linguistics*, pages 193–200.

Michael Brent. 1993. From grammar to lexicon: Unsupervised learning of lexical syntax. *Computational Linguistics*, 19(2):243–262.

Ted Briscoe and John Carroll. 1997. Automatic extraction of subcategorization from corpora. In *Proceedings of the 5th Conference on Applied Natural Language Processing*, pages 356–363.

Glenn Carroll and Mats Rooth. 1998. Valence induction with a head-lexicalized PCFG. In *Proceedings of the 3rd Conference on Empirical Methods in Natural Language Processing*, pages 36–45.

Murat Ersan and Eugene Charniak, 1996. *A Statistical Syntactic Disambiguation Program and What It Learns*, pages 146–157. Springer.

Susanne Gahl. 1998. Automatic extraction of subcorpora based on subcategorization frames from a part-of-speech tagged corpus. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*, pages 428–432.

Ralph Grishman, Catherine Macleod, and Adam Meyers. 1994. COMLEX Syntax: Building a computational lexicon. In *Proceedings of COLING1994*, pages 268–272.

Daisuke Kawahara and Sadao Kurohashi. 2005. Pp-attachment disambiguation boosted by a gigantic volume of unambiguous examples. In *Proceedings of IJCNLP-05*, pages 188–198.

Daisuke Kawahara and Sadao Kurohashi. 2006. Case frame compilation from the web using high-performance computing. In *Proceedings of LREC2006*, pages 1344–1347.

Daisuke Kawahara and Kiyotaka Uchimoto. 2008. A method for automatically constructing case frames for English. In *Proceedings of LREC2008*.

Anna Korhonen and Judita Preiss. 2003. Improving subcategorization acquisition using word sense disambiguation. In *Proceedings of ACL2003*, pages 48–55.

Anna Korhonen, Yuval Krymolowski, and Ted Briscoe. 2006. A large subcategorization lexicon for natural language processing applications. In *Proceedings of LREC2006*.

Anna Korhonen. 2002. Semantically motivated subcategorization acquisition. In *Proceedings of the ACL-02 Work-*

shop on Unsupervised Lexical Acquisition, pages 51–58.

Taku Kudo and Yuji Matsumoto. 2001. Chunking with support vector machines. In *Proceedings of the 2nd Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 192–199.

Maria Lapata. 1999. Acquiring lexical generalizations from corpora: A case study for diathesis alternations. In *Proceedings of ACL1999*, pages 397–404.

Christopher Manning and Hinrich Schütze. 1999. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, Massachusetts.

Christopher Manning. 1993. Automatic acquisition of a large subcategorization dictionary from corpora. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, pages 235–242.

Mitchell Marcus, Beatrice Santorini, and Mary Marcinkiewicz. 1994. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.

Sujith Ravi, Kevin Knight, and Radu Soricut. 2008. Automatic prediction of parser accuracy. In *Proceedings of EMNLP2008*, pages 887–896.

Roi Reichart and Ari Rappoport. 2007. An ensemble method for selection of high quality parses. In *Proceedings of ACL2007*, pages 408–415.

The XTAG Research Group, editor. 1998. *A Lexicalized Tree Adjoining Grammar for English*.

Yoshimasa Tsuruoka and Jun'ichi Tsujii. 2005. Bidirectional inference with the easiest-first strategy for tagging sequence data. In *Proceedings of HLT-EMNLP2005*, pages 467–474.

Akira Ushioda, David Evans, Ted Gibson, and Alex Waibel. 1993. The automatic acquisition of frequencies of verb subcategorization frames from tagged corpora. In *Proceedings of the Workshop on Acquisition of Lexical Knowledge from Text*, pages 95–106.