A Hybrid Approach to Deriving Selectional Preferences

Arendse Bernth and **Michael C. McCord**
IBM T.J. Watson Research Center
P.O. Box 218
Yorktown Heights, NY 10598
{arendse,mcmccord}@us.ibm.com

Abstract

A hybrid approach to automatic derivation of class-based selectional preferences is proposed. A lexicon of selectional preferences can assist in handling several forms of ambiguity, a major problem for MT. The approach combines knowledge-rich parsing and lexicons, with statistics and corpus data. We illustrate the use of a selectional preference lexicon for anaphora resolution.

1 Introduction

In this paper we propose a hybrid approach to automatic derivation of selectional preferences. Selectional preferences characterize the potential arguments of word senses in terms of their semantic properties (Resnik, 1998). An oft-quoted example is that the verb *eat* strongly prefers an object in the category of *food*, so much so that one can omit the object without causing confusion (Levin, 1993; Resnik, 1996).

A lexicon of selectional preferences can assist in handling *ambiguity*, a major problem for MT, be it semantic or structural. For example, selectional preferences aid in pronoun resolution (Bernth, 2002) and word sense disambiguation (Resnik, 1997). Selectional preferences can also aid parsing by rewarding parses that have more "natural" arguments for words. Finally, selectional preferences can be used to infer semantic properties of words missing from the lexicon.

Rational methods have often been criticized for being labor-intensive, inflexible, and hard to scale up, but praised for being deep, accurate and information-rich. Empirical methods have been criticized for being inaccurate, simple-minded, and domain-specific, and praised for being automatic and providing good coverage.

Hybrid approaches aim at maximizing the benefits, while minimizing the disadvantages, of each approach, and popularity of hybrid systems is evidenced by papers such as (Carl et al., 2002) and (Habash and Dorr, 2002).

In Section 2 we describe the rational and empirical components of our system, and how they are combined into a hybrid system for deriving selectional constraints. Section 3 describes experiments and results.

2 Resources and Methods

Derivation of selectional preferences seems like a particularly good candidate for a hybrid approach. On the one hand, it is imperative to get a precise indication of syntactic dependencies, obviously a reasonable job for a meticulous parser. And on the other hand, it is important to acquire the actual preferences by gathering evidence from real data; this is obviously the empirical approach. The system that we propose combines rational and empirical components: Knowledge-rich parsing and lexicons, combined with statistics and corpus data. In Section 2.1 we describe the rational components, and in Section 2.2 we describe the empirical components. In Section 2.3 we report on the specific benefits of this combined approach.

2.1 The Rational Components

The rational components consist of the parser, described in Section 2.1.1, and the lexicon and semantic type hierarchy described in Section 2.1.2.

2.1.1 The Parser

English Slot Grammar (ESG), a broad-scale, general English parsing environment (McCord, 1980; McCord, 1990; McCord, 1993), provides the core of the rational aspect. ESG handles a variety of text formats, such as HTML, SGML, and plain text. The ESG system segments and tokenizes the input text, performs morphological analysis (including derivational as well as inflectional morphology), and finally assigns syntactic structures to the sentences. The syntactic structures show not only surface relations, but also deeper relations, as exemplified by logical argument analysis for passive constructions, coordination, extraposition, and VPs without overt subjects. (Firth, 1957) says: “You shall know a word by the company it keeps.” Even though this aphorism was not uttered in the context of computational parsing, and is often quoted in the justification of empirical methods, it also seems appropriate in the context of a rational, deep parsing system: A full, information-rich parse gives a very good indication of what company a word keeps, better than what is provided by near-neighbor n-gram methods, because the most important “company” information for words is is in their modifier or *slot-filler* relationships, which may be remote in the sentence.

Central to Slot Grammar is the concept of a *slot*. A slot is a grammatical function, like subject, object or indirect object, but there are many slots in Slot Grammar. Slots are either *complement slots* or *adjunct slots*. Complement slots are associated with each word sense in the lexicon, in a list called the *slot frame* for the word sense. Complement slots can also be viewed as names for the arguments of a word sense viewed as a predicate in logical form. Adjunct slots are associated with parts of speech in the grammar. All open-class words (verbs, nouns, adjectives, adverbs) can have complement slots, but we will be concerned only with gathering slot frame data for verbs in the work reported on here. Slot Grammar uses only six complement slots for verbs (there are many more adjunct slots): `subj` (subject), `obj` (direct object), `iobj` (indirect object), `comp` (complement of object or subject), `auxcomp` (auxiliary complement), and `pred` (predicate complement of “be”). But slots can have several different *(slot) options*, which determine the lexical category of the filler. For instance `iobj` can have the option `n` for an NP filler or the option `to` for a *to*-PP filler. The general idea is that a given complement slot represents a particular argument of a word sense predicate, and the slot’s various options represent how that argument can be realized morphosyntactically. The specific slot frames for word senses in the lexicon show which options are allowed for each slot for that word sense.

Figure 1 shows an example of an ESG parse, including the semantic types applicable for the individual words. For example, *caviar* is marked with the semantic type *st_food*, and *vodka* has the semantic type *st_liquid*.

2.1.2 The Lexicon and Semantic Types

ESG uses a broad-coverage lexicon with word senses marked with semantic types that are organized in an isa hierarchy. The lexicon has approximately 94,000 base forms, with many more word forms covered by inflectional and derivational morphology. Slot frames are generally well-marked in the lexicon. The lexical system allows for multiwords, and addendum multiword lexicons can usefully include named entities. The semantic type hierarchy has approximately 450 types, and is hand-coded, based on our lexical development for the LMT machine translation system (McCord and Bernth, 1998). This type hierarchy is quite small compared with an ontology like WordNet (Fellbaum, 1998) (though it will probably expand), but our general idea is to use a relatively small set of semantic types to *distinguish* word senses as necessary in our systems, as opposed to defining senses.

2.2 The Empirical Components

The empirical components comprise large-scale corpora as described in Section 2.2.1 and frequency counts and maximum likelihood estimation, discussed in Section 2.2.2.

2.2.1 Robust Corpus Processing

We have used a corpus of unannotated Reuters newswire comprising approximately 6.4 million sentences. In order to robustly handle this amount of data, we employed some techniques from our terminology extraction tool described in (Bernth et al., 2002), and from (McCord, 1993). On top of ESG we

```
------------------------------------------------------------------
  .- ndet    the1(1)         det pl def the ingdet (def the)
  .- nadj    Russian1(2)     adj (hlanguage st_people)
.--- subj(n) emperor1(3)     noun cn pl title (title m)
.--- lconj   eat1(4,3,5,u)   verb vfin vpast sg pl
| ‘- obj(n)  caviar1(5)      noun cn sg (massn st_food)
o--- top     and0(6)         verb vfin vpast pl vsubj
‘--- rconj   drink1(7,3,8,u) verb vfin vpast sg pl
  ‘- obj(n)  vodka1(8)       noun cn sg (st_liquid)
------------------------------------------------------------------
```

Figure 1: ESG parse of “The Russian emperors ate caviar and drank vodka”

have programs that can extract filled slot frame data from parses and accumulate unique frames, along with their frequencies, robustly across vast amounts of text data. In Section 2.3 we describe the frames in more detail.

The corpus processing can operate on multiple files specified by file patterns, or on lists of file names (or file patterns), or even by web crawling. The system can gather such frames for any part of speech, but in the scenario described here, we are only concerned with verbs and their frames. We are focusing on verbs because their filled frames are more useful for anaphora resolution than those of other parts of speech. But our methodology applies to any parts of speech.

2.2.2 Frequency and Maximum-Likelihood Estimation

A variation on simple relative frequency determines the selectional preferences for complements. Let the frequency of a specific slot frame f for a verb v in the training corpus be $freq(f)$. The following then describes the simple relative frequency of a specific slot frame f_0:

$$\frac{freq(f_0)}{\sum_{f \in F} freq(f)}$$

where F is the set of frames for v.

This maximum likelihood estimate assigns zero probability to unseen events, a well-known problem causing undesirable results for sparse data. This is quite similar to the idea of negation-as-failure, probably best known from the programming language Prolog (Clocksin and Mellish, 1981). In both cases the problem can be traced down to the closed-world assumption, which, simply stated, is the assumption that our information is complete, be it the training data or the Prolog database. This assumption is sufficient for many cases and leads to increased efficiency, but does also cause unknown/unseen cases to just not be considered legitimate. However, our lexicon is broad-coverage, and rather complete in terms of slot frames (as indicated in Section 2.1.2), and our corpus big, so we allow ourselves to take advantage of the simplicity of MLE.

2.3 Advantages of the Hybrid Approach

Prior approaches such as (Dagan and Itai, 1990), (Resnik, 1993), (Li and Abe, 1996) and (Li and Abe, 1998) limit themselves to *single* grammatical relations – individual arguments that a verb may take – considered independently of the total slot frame. However, as (Li and Abe, 1996) correctly points out, considering full slot frames rather than just single relations will give better and more accurate results in certain cases. For instance, suppose we are resolving the pronoun in *The cow ate it*, and possible antecedents are *mouse* and *grass*. Both mice and grass can be eaten (by suitable animals), so storing the possible direct objects or semantic types of objects for the verb *eat* will not help in the resolution. But storing subject-verb-object frames can tell us that cows eat grass but cows don’t typically eat mice. Unlike (Li and Abe, 1996) we are able to gather complete frames because we are not limited to a relatively small training corpus such as the Penn Treebank.[1] They were only able to train on

[1](Li and Abe, 1998), which presumably reports on the same project, indicates that the training corpus comprised 126,084 sentences of tagged text from the *Wall Street Journal*.

little more than 125,000 sentences because they had to rely on a human-annotated training corpus. However, ESG provides us with high-quality parses, and what this entails in practice is that we are able to train on a virtually unlimited amount of data and use high-frequency frames to obtain preferences.

Figure 2 shows sample slot frame output from our experiment, for the verb *eat*, with frequency information. In general, each line of slot frame output shows a filled slot frame followed by the symbol `<` and the frequency for that slot frame. A filled slot frame is a list of terms of one of the two forms:

$$(Slot\ Option\ Type_1\ Type_2 \ldots)$$
$$(Slot\ \texttt{u})$$

Here $Slot$ is an ESG slot. The verb slots `subj`, `obj`, and `comp` are shown in the figure. The $Option$ is the chosen slot option for $Slot$ as described above in Section 2.1.1. The option `n`, used in the figure, indicates a nominal phrase filler. Each $Type_i$ names a semantic type of the filler. In the figure, semantic types `h` (human) and `st_food` (food) are shown. The symbol `u` (for "unfilled" or "unknown") in the position of the $Option$ indicates that a slot is not filled.

Our first version of the system, briefly reported on in (Bernth, 2002), followed a word-based approach similar to (Dagan and Itai, 1990) in that actual *words* filling the slots were harvested for the slot frames. This approach suffers from the drawback of producing lower-frequency results for complete slot frames since the combinations of actual words for a given verb are not likely to occur so frequently. In order to further increase the useful frequencies, we chose to follow the approach of e.g. (Resnik, 1993) and (Li and Abe, 1998), using *class-based* models to generalize the results. Class-based models assign probability values to classes of words rather than to individual words.

Our word classes are just the sets of words that have particular bundles of semantic types from our type hierarchy. But we chose to conflate the type bundles (and hence the associated word classes) by *raising* certain semantic types to a selected set of "super semantic types". For each super semantic type T, any semantic type that is below T in the type hierarchy is replaced by T. For example, Human is one of our chosen super semantic types, and a lower semantic type such as Artist, if it occurs, will be replaced by Human. The reasoning is that the super semantic types make distinctions enough of the time for selectional preferences. And this conflation of word classes increases the useful frequencies of frames.

3 Experiments and Results

We applied our method to approximately 6.4 million sentences from a corpus of Reuters newswire, resulting in slot frames for 6760 verbs; this accounts for approximately 75 percent of all verbs in our lexicon. Using an option to output only entries with frequencies higher than a given threshold, we removed low-frequency slot frames and slot frames for which no semantic types were available. An unexpected number of slot frames had to be removed for e.g. verbs that take *that*-complements because we do not yet take into account the semantic type of the head of the embedded clause; deciding on the proper handling of this is a nontrivial problem that we will return to. Additionally, the lexicon that is available to us is not completely marked up with semantic types.

3.1 Qualitative Results

We compared the results of our system with the list given in (Resnik, 1996). A major difference is that the results reported in (Resnik, 1996) only relate an *object* to a verb, not the complete slot frame. Needless to say, this makes the comparison harder, since we are comparing our more complete slot frames to Resnik's *partial* slot frames. However, it is still valuable to make a comparison and determine to what extent the two systems agree on the verb-object relation, and to give additional results pertaining to a fuller verb frame.

The results are displayed in Table 1. The **Assoc** and **WN Class** are the values given in (Resnik, 1996), and the **SelPref** (selectional preference) and **SG Class** columns refer to our system. The **Assoc** value indicates the degree to which a verb prefers (or disprefers) an object of the **WN Class** (WordNet class). It is computed based on relative entropy. The values given for our system in the **SelPref** column are MLEs and hence range from zero to 1. The last column gives the frequency for a given frame followed in parentheses by the total number of verb

```
eat
   (subj n h)(obj n st_food)(comp u)) < 51
   (subj u)(obj n st_food)(comp u))   < 35
```

Figure 2: Sample slot frame output for the verb *eat*

frames for that verb that have both subject and object slot filled.[2]

There is not a complete match between the semantic types from WordNet that (Resnik, 1996) uses and our semantic types; this obviously makes a comparison harder. For several verbs we found it informative to give more than one frame. We also found it informative to show results for some verbs that are not listed in (Resnik, 1996).

As can be seen from Table 1, there is clear agreement in the semantic classes in most of the cases. Differences occur for e.g. *see* where (Resnik, 1996) conflates the object class at a higher level (and probably correctly so). Even so, no one can argue that humans and documents are not valid object classes for *see*. Generally speaking, (Resnik, 1996) conflates classes at a higher level than we do. For this corpus, there is a high propensity for human subjects, and this may be caused by the fact that we conflate several semantic types under Human, e.g. *st_company* and *st_community*. In the list of additional verbs we have included some verb frames with non-human subjects as well as frames with human subjects. We will experiment more with less conflation and more data.

3.2 Applying the Selectional Preferences

Anaphora resolution is an obvious candidate for applying selectional preferences, and in fact the main motivation for the present work. The importance of anaphora resolution for MT has long been recognized, for instance in producing the right gender and semantic types for translations of pronouns.

There are a variety of approaches to anaphora resolution, but most systems agree on the importance of morphological agreement, recency, identical surface grammatical role, and frequency of particular possible antecedents occurring in the text (Mitkov, 2002).

[2] In some cases we have given *all* slot frames for a given verb; in other cases just one or more examples. Hence the absolute number of occurrences stated for the slot frames may or may not add up to the number in parentheses.

Whereas these certainly are useful, there are also a number of cases where they are not enough.

(1) The food was put on the table by the cook.
He then sat down to eat it.

Applying morphological agreement to resolution of the pronoun *it* in (1) leaves us with two candidates, *food* and *table*. Applying the rule of recency, a resolution algorithm would choose the wrong candidate *table*. Likewise, applying the rule of identical surface role will not resolve the pronoun. If the two sentences of this example were part of a larger context, then antecedent frequency might, or might not, contribute something.

However, it is very clear to humans that the antecedent of *it* is *food*. This is due to the selectional preferences for *eat*.

The Euphoria anaphora resolution system (Bernth, 2002) uses semantic type checking and certain syntactic constraints, in addition to the above-mentioned common rules, but was unable to correctly resolve the reference in example (1). However, after adding the total derived lexicon of selectional preferences to Euphoria and integrating its use, the reference was correctly resolved. We will report separately on a more extensive quantitative evaluation of the improvement in performance for anaphora resolution.

4 Conclusion

We have reported on a large-scale hybrid system for automatically acquiring selectional preferences. The system utilizes a combination of a full-fledged, broad-coverage parser and statistical measures to acquire full slot frames for verbs with semantic classes for the arguments. The hybrid approach allows us to train on a virtually unlimited amount of data, and gives high precision combined with broad coverage. By extracting slot frames from a large corpus in a newswire domain, we have acquired selectional preferences in that domain that cover about 75 percent of the verbs in a commercially used general-purpose dictionary. Finally we have illustrated the use of the acquired selectional preference lexicon for anaphora resolution.

References

Arendse Bernth, Michael McCord, and Kara Warburton. 2002. Terminology extraction for global content management. Technical Report RC22615, IBM T.J. Watson Research Center, P.O. Box 218, Yorktown Heights, NY 10598, November. To appear in *Terminology*.

Arendse Bernth. 2002. Euphoria – A reference resolution system for machine translation. Technical Report RC22627, IBM T.J. Watson Research Center, P.O. Box 218, Yorktown Heights, NY 10598, November.

Michael Carl, Andy Way, and Reinhard Schäler. 2002. Toward a hybrid integrated translation environment. In Stephen D. Richardson, editor, *Machine Translation: From Research to Real Users, 5th Conference of the Association for Machine Translation in the Americas*, number 2499 in Springer Lecture Notes in Artificial Intelligence, pages 11–20, Tiburon, CA, USA. AMTA.

William F. Clocksin and Christopher S. Mellish. 1981. *Programming in Prolog*. Springer-Verlag, Berlin, Heidelberg, New York.

Ido Dagan and Alon Itai. 1990. Automatic acquisition of constraints for the resolution of anaphora references and syntactic ambiguities. In *Proceedings of COLING-90*, volume 3, pages 330–332, Helsinki, Finland.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Database*. MIT Press, Cambridge, Massachusetts.

J. R. Firth. 1957. A synopsis of linguistic theory 1930–1955. In *Studies in Linguistic Analysis*, pages 1–32. Philological Society, Oxford.

Nizar Habash and Bonnie Dorr. 2002. Handling translation divergences: Combining statistical and symbolic techniques in generation-heavy machine translation. In Stephen D. Richardson, editor, *Machine Translation: From Research to Real Users, 5th Conference of the Association for Machine Translation in the Americas*, number 2499 in Springer Lecture Notes in Artificial Intelligence, pages 84–93, Tiburon, CA, USA. AMTA.

Beth Levin. 1993. *English Verb Classes and Alternations*. University of Chicago Press, Chicago.

Hang Li and Naoki Abe. 1996. Learning dependencies between case frame slots. In *Proceedings of the Sixteenth International Conference on Computational Linguistics*, pages 10–15, Copenhagen, Denmark.

Hang Li and Naoki Abe. 1998. Generalizing case frames using a thesaurus and the MDL principle. *Computational Linguistics*, 24(2):217–244.

Michael C. McCord and Arendse Bernth. 1998. The LMT transformational system. In *Proceedings of AMTA-98*, pages 344–355. Association for Machine Translation in the Americas.

Michael C. McCord. 1980. Slot Grammars. *Computational Linguistics*, 6:31–43.

Michael C. McCord. 1990. Slot Grammar: A system for simpler construction of practical natural language grammars. In R. Studer, editor, *Natural Language and Logic: International Scientific Symposium, Lecture Notes in Computer Science*, pages 118–145. Springer Verlag, Berlin.

Michael C. McCord. 1993. Heuristics for broad-coverage natural language parsing. In *Proceedings of the ARPA Human Language Technology Workshop*, pages 127–132, Princeton, New Jersey. Morgan-Kaufmann.

Ruslan Mitkov, editor. 2002. *Anaphora Resolution*. Longman, London.

Philip S. Resnik. 1993. *Selection and Information: A Class-Based Approach to Lexical Relationships*. Ph.D. thesis, University of Pennsylvania.

Philip Resnik. 1996. Selectional constraints: An information-theoretic model and its computational realization. *Cognition*, 61:127–159.

Philip Resnik. 1997. Selectional preference and sense disambiguation. In *Proceedings of the ACL SIGLEX Workshop on Tagging Text with Lexical Semantics: What, Why, and How?*, pages 52–57, Washington, DC. Association for Computational Linguistics.

Philip Resnik. 1998. WordNet and class-based probabilities. In Christiane Fellbaum, editor, *WordNet: An Electronic Database*, pages 239–263. MIT Press, Cambridge, Massachusetts.

Table 1: Comparison

Verb	**Assoc** *object*	**WN Class** *object*	**SG Class** *object*	**SG Class** *subject*	**SelPref**	**#**
see	5.79	<entity>	<human>	<human>	0.468	22 (47)
			<st_document>	<human>	0.277	13 (47)
read	6.80	<writing>	<st_document>	<human>	0.624	73 (117)
			<speech_act>	<human>	0.377	44 (117)
hear	1.89	<communication>	<st_document>	<human>	0.212	41 (193)
			<speech_act>	<human>	0.192	37 (193)
			<st_info>	<human>	0.088	17 (193)
			<human>	<human>	0.249	48 (193)
write	7.26	<writing>	<st_document>	<human>	0.950	132 (139)
urge	1.14	<life form>	<human>	<human>	0.938	2340 (2496)
warn	4.73	<person>	<human>	<human>	1.000	79 (79)
judge	1.30	<contest>	<human>	<human>	0.524	11 (21)
			<st_interaction>	<human>	0.476	10 (21)
teach	1.87	<cognition>	<st_discipline>	<human>	0.210	21 (60)
			<st_document>	<human>	0.083	5 (60)
			<human>	<human>	0.417	25 (60)
expect	0.59	<act>	<human>	<human>	0.366	26 (71)
repeat	1.23	<communication>	<speech_act>	<human>	0.582	32 (55)
			<st_document>	<human>	0.343	12 (55)
understand	1.52	<cognition>	<st_cognition>	<human>	0.159	10 (63)
			<st_interaction>	<human>	0.238	15 (63)
			<st_problem>	<human>	0.222	14 (63)
			<st_document>	<human>	0.190	12 (63)
			<st_need>	<human>	0.190	12 (63)
Not in Resnik's list:						
measure			<st_outcome>	<st_document>	0.464	150 (323)
eat			<st_food>	<human>	0.746	135 (181)
drink			<st_liquid>	<human>	0.882	60 (68)
kill			<human>	<human>	0.869	2109 (2428)
			<human>	<st_event>	0.036	87 (2428)
			<human>	<air_vehicle>	0.028	68 (2428)
			<human>	<st_weapon>	0.012	29 (2428)
			<human>	<st_animal>	0.004	10 (2428)
love			<human>	<human>	0.855	106 (124)
			<st_place>	<human>	0.145	18 (124)
throw			<st_event>	<human>	0.576	19 (33)
			<st_artifact>	<human>	0.424	14 (33)
describe			<human>	<human>	0.722	65 (90)
			<st_event>	<human>	0.156	14 (90)
			<st_document>	<human>	0.122	11 (90)
study			<st_document>	<human>	0.534	119 (223)
			<st_cognition>	<human>	0.224	50 (223)
			<st_action>	<human>	0.049	11 (223)
attack			<human>	<human>	0.671	496 (739)
			<st_place>	<human>	0.099	73 (739)