Empirical Acquisition of Differentiating Relations from Definitions

Tom O'Hara
Department of Computer Science
New Mexico State University
Las Cruces, NM 88003
tomohara@cs.nmsu.edu

Janyce Wiebe
Department of Computer Science
University of Pittsburgh
Pittsburgh, PA 15260
wiebe@cs.pitt.edu

Abstract

This paper describes a new automatic approach for extracting conceptual distinctions from dictionary definitions. A broad-coverage dependency parser is first used to extract the lexical relations from the definitions. Then the relations are disambiguated using associations learned from tagged corpora. This contrasts with earlier approaches using manually developed rules for disambiguation.

1 Introduction

Large-scale lexicons for computational semantics often lack sufficient distinguishing information for the concepts serving to define words. For example, WordNet (Miller, 1990) recently introduced new relations for domain category and location in Version 2.0, along with 6,000+ instances; however, about 38% of the noun synsets are still not explicitly distinguished from sibling synsets.

Work on the Extended WordNet project (Harabagiu et al., 1999) is achieving substantial progress in making the information in WordNet more explicit. The main goal is to transform the definitions into a logical form representation suitable for drawing inferences; in addition, the content words in the definitions are being disambiguated. In the logical form representation, separate predicates are used for each preposition, as well as for some other functional words (e.g., conjunctions); thus, ambiguity in the underlying relations implicit in the definitions is not being resolved. The work described here automates the process of relation disambiguation. This can be used to further the transformation of WordNet into an explicit lexical knowledge base.

Earlier approaches to differentia extraction have predominantly relied upon manually constructed pattern matching rules for extracting relations from dictionary definitions (Vanderwende, 1996; Barrière, 1997; Rus, 2002). These rules can be very precise, but achieving broad-coverage can be difficult. Here a broad coverage dependency parser is first used to determine the syntactic relations that are present among the constituents in the sentence. Then the syntactic relations between sentential constituents are converted into semantic relations between the underlying concepts using statistical classification.

Isolating the disambiguation step from the extraction step in this manner allows for greater flexibility over earlier approaches. For example, different parsers can be incorporated without having to rework the disambiguation process.

This paper is organized as follows: Section 2 details the steps in extracting the initial relations from the definition parse. Section 3 illustrates the disambiguation process, the crucial part of this approach. Section 4 presents an evaluation of the relations that are extracted from the WordNet definitions. Lastly, Section 5 compares the approach to previous approaches that have been tried.

2 Differentia Extraction

The approach to differentia extraction is entirely automated. This starts with using the *Link Grammar Parser* (Sleator and Temperley, 1993), a dependency parser, to determine the syntactic lexical relations that occur in the sentence. Dictionary definitions are often given in the form of sentence fragments with the headword omitted. For example, the definition for the beverage sense of 'wine' is "fermented juice (of grapes especially)." Therefore, prior to running the definition analysis, the definitions are converted into complete sentences, using simple templates for each part of speech.

After parsing, a series of postprocessing steps is performed prior to the extraction of the lexical relations. For the Link Parser, this mainly involves conversion of the binary dependencies into relational tuples and the realignment of the tuples around function words. The Link Parser outputs syntactic dependencies among words, punctuation, and sentence boundary markers. The parser uses quite specialized syntactic relations, so these are converted into general ones prior to the extraction of the relational tuples. For example, the relation A, which is used for pre-noun adjectives, is converted into *modifies*. Figure 1 illustrates the syntactic relations that would be extracted, along with the original parser output.

The syntactic relationships are first converted into relational tuples using the format ⟨source-word, *relation-word*, target-word⟩. This conversion is performed by following the dependencies involving the content words, ignoring cases involving non-word elements (e.g., punctuation). For example, the first tuple extracted from the parse would be ⟨n:wine,

Definition sentence:
    Wine is fermented juice (of grapes especially).

Link Grammar parse:
    ⟨/////, Wd, 1. n:wine⟩
    ⟨/////, Xp, 10. .⟩
    ⟨1. n:wine, Ss, 2. v:is⟩
    ⟨10. ., RW, 11. /////⟩
    ⟨2. v:is, Ost, 4. n:juice⟩
    ⟨3. v:fermented, A, 4. n:juice⟩
    ⟨4. n:juice, MXs, 6. of⟩
    ⟨5. (, Xd, 6. of⟩
    ⟨6. of, Jp, 7. n:grapes⟩
    ⟨6. of, Xc, 9.)⟩

Extracted relations:
    ⟨1. n:wine, 2. v:is, 4. n:juice⟩
    ⟨3. v:fermented, modifies-3-4, 4. n:juice⟩
    ⟨4. n:juice, 6. of, 7. n:grapes⟩

Figure 1: *Example for relation extraction.*

v:is, n:juice⟩. Certain types of dependencies are treated specially by converting the syntactic relationships directly into a relational tuple involving a special relation-indicating word (e.g., 'modifies').

The relational tuples extracted from the parse form the basis for the lexical relations derived from the definition. Structural ambiguity resolution is not addressed here, so the first parse returned is used. The remaining optional step assigns weights to the relations that are extracted.

When using the relations in applications, it is desirable to have a measure of how relevant the relations are to the associated concepts. One such measure would be the degree to which the relation applies to the concept being described as opposed to sibling concepts. To account for this, *cue validities* are used, borrowing from cognitive psychology (Smith and Medin, 1981). Cue validities can be interpreted as probabilities indicating the degree to which features apply to a given concept versus similar concepts (i.e., $P(C|F)$).

Cue validities are estimated by calculating the percentage of times that the feature is associated with a concept versus the total associations of contrasting concepts. This requires a means of determining the set of contrasting concepts for a given concept. The simplest way of doing this would be to just select the set of sibling concepts (e.g., synsets sharing a common parent in WordNet). However, due to the idiosyncratic way concepts are specialized in knowledge bases, this likely would not include concepts intuitively considered as contrasting.

To alleviate this problem the *most-informative ancestor* will be used instead of the parent. This is determined by selecting the ancestor that best balances frequency of occurrence in a tagged corpus with specificity. This is similar to Resnik's (1995) notion of most-informative subsumer for a pair of

concepts. In his approach, estimated frequencies for synsets are percolated up the hierarchy, so that the frequency always increases as one proceeds up the hierarchy. Therefore the first common ancestor for a pair is the most-informative subsumer (i.e., has most information content). Here attested frequencies from SemCor (Miller et al., 1994) are used, so all ancestors are considered. Specificity is accounted for by applying a scaling factor to the frequencies that decreases as one proceeds up the hierarchy. Thus, 'informative' is used more in an intuitive sense rather than technical.

More details on the extraction process and the subsequent disambiguation can be found in (O'Hara, forthcoming).

3 Differentia Disambiguation

After the differentia properties have been extracted from a definition, the words for the relation source and object terms are disambiguated to order to reduce vagueness in the relationships. In addition, the relation types are converted from surface-level relations (e.g., object) or relation-indicating words (e.g., prepositions) into the underlying semantic relation.

Since WordNet serves as the knowledge base being targeted, term disambiguation involves selecting the most appropriate synset for both the source and target terms. The WordNet definitions have recently been sense-tagged as part of the Extended WordNet (Novischi, 2002), so these annotations are incorporated. For other dictionaries, use of traditional word-sense disambiguation algorithms would be required.

With the emphasis on corpus analysis in computational linguistics, there has been shift away from relying on explicitly coded knowledge towards the use of knowledge inferred from naturally occurring text, in particular text that has been annotated by humans to indicate phenomena of interest. The PENN TREEBANK version II (Marcus et al., 1994) provided the first large-scale set of case role annotations for general-purpose text. These are very general roles akin to Fillmore's (1968) case roles. The Berkeley FRAMENET (Fillmore et al., 2001) project provides the most recent large-scale annotation of semantic roles. These are at a much finer granularity than those in Treebank, so they should prove quite useful for applications learning detailed semantics from corpora. O'Hara and Wiebe (2003) explain how both inventories can be used for preposition disambiguation.

The goal of relation disambiguation is to determine the underlying semantic role indicated by particular words in a phrase or by word order. For relations indicated directly by function words, the disambiguation can be seen as a special case of word-sense disambiguation (WSD). As an example, refining the relationship ⟨'dog', *'with'*, 'ears'⟩ into ⟨'dog', *has-part*, 'ears'⟩, is equivalent to disambiguating the preposition 'with,' given that the senses are the dif-

Local-context features

POS:	part of speech of target word
POS$-i$:	part-of-speech of ith word to left
POS$+i$:	part-of-speech of ith word to right
Word:	target wordform as is
Word$-i$:	ith word to the left
Word$+i$:	ith word to the right

Collocational features

WordColl$_i$:	word collocation for sense i

Class-based collocational features

HyperColl$_s$:	hypernym collocation for sense i

Figure 2: **Features for preposition classifier.**

ferent relations it can indicate. For relations that are indicated implicitly (e.g., adjectival modification), other classification techniques would be required, reflecting the more syntactic nature of the task.

A straightforward approach for preposition disambiguation would be to use standard WSD features, such as the parts-of-speech of surrounding words and, more importantly, collocations (e.g., lexical associations). Although this can be highly accurate, it tends to overfit the data and to generalize poorly. The latter is of particular concern here as the training data is taken from a different genre (e.g., newspaper text rather than dictionary definitions). To overcome these problems, a class-based approach is used for the collocations, with WordNet high-level synsets as the source of the word classes. Figure 2 lists the features used for the classifier.

For the application to differentia disambiguation, the classifiers learned over Treebank and FrameNet need to be combined. This can be done readily in a cascaded fashion with the classifier for the most specific relation inventory (i.e., FrameNet) being used first and then the other classifiers being applied in turn whenever the classification is inconclusive. This has the advantage that new resources can be integrated into the combined relation classifier with minimal effort. However, the resulting role inventory will likely be heterogeneous and might be prone to inconsistent classifications. In addition, the role inventory could change whenever new annotation resources are incorporated, making the differentia disambiguation system less predictable.

Alternatively, the annotations can be converted into a common inventory, and a separate relation classifier induced over the resulting data. This has the advantage that the target relation-type inventory remains stable whenever new sources of relation annotations are introduced. The drawback however is that annotations from new resources must first be mapped into the common inventory before incorporation. The latter approach is employed here. The common inventory incorporates some of the general relation types defined by Gildea and Jurafsky (2002) for their experiments in classifying semantic relations in FrameNet using a reduced inventory.

Relation	Frequency
Theme	0.316
Goal	0.116
Ground	0.080
Category	0.069
Agent	0.069
Cause	0.061
Manner	0.058
Recipient	0.053
Medium	0.039
Characteristic	0.022
Resource	0.021
Means	0.021
Source	0.019
Path	0.017
Experiencer	0.017
Accompaniment	0.011
Area	0.010
Direction	0.001

Table 1: **Frequency of relations extracted.**

4 Evaluation

The evaluation discussed here assesses the quality of the information that would be added to the lexicons with respect to relation disambiguation, which is the focus of the research. An application-oriented evaluation is discussed in (O'Hara, forthcoming), showing how using the extracted information improves word-sense disambiguation.

All the definitions from WordNet 1.7.1 were run through the differentia-extraction process. This involved 111,223 synsets, of which 10,810 had preprocessing or parse-related errors leading to no relations being extracted. Table 1 shows the frequency of the relations in the output from the differentia extraction process. The most common relation used is *Theme*, which occurs four times as much compared to the annotations. It is usually annotated as the sense for 'of,' which also occurs with roles *Source*, *Category*, *Ground*, *Agent*, *Characteristic*, and *Experiencer*. Some of these represent subtle distinctions, so it is likely that the difference in the text genre is causing the classifier to use the default more often.

Four human judges were recruited to evaluate random samples of the relations that were extracted. To allow for inter-coder reliability analysis, each evaluator evaluated some samples that were also evaluated by the others, half as part of a training phase and half after training. In addition, they also evaluated a few samples that were manually corrected beforehand. This provides a baseline against which the uncorrected results can be measured against. Because the research only addresses relations indicated by prepositional phrases, the evaluation is restricted to these cases. Specifically, the judges rate the assignment of relations to the prepositional phrases on a scale from 1 to 5, with 5 being an exact match.

The evaluation is based on averaging the assess-

Corrected			Uncorrected	
#Cases	10		#Cases	15
#Scores	40		#Scores	60
Mean	3.225		Mean	3.033
STDEV	1.625		STDEV	1.551
Score	0.60		Score	0.58

Table 2: **Mean assessment score for all extracted relationships.** 25 relationships were each evaluated by 4 judges. *Mean* gives the mean of the assessment ratings (from 1 to 5). *Score* gives ratings relative to scale from 0 to 1.

ment scores over the relationships. Table 2 shows the results from this evaluation, including the manually corrected as well as the uncorrected subsets of the relationships. For the corrected output, the mean assessment value was 3.225, which translates into an overall score of 0.60. For the uncorrected system output, the mean assessment value was 3.033, which translates into an overall score of 0.58. Although the absolute score is not high, the system's output is generally acceptable, as the score for the uncorrected set of relationships is close to that of the manually corrected set.

5 Related work

Most of the work addressing differentia extraction has relied upon manually constructed extraction rules (Vanderwende, 1996; Barrière, 1997; Rus, 2002). Here the emphasis is switched from transformation patterns for extracting relations into statistical classification for relation disambiguation, given tagged corpora with examples. This allows for better coverage at the expense of precision. Note that relation disambiguation is not yet addressed in Extended WordNet (Rus, 2002); for example, prepositions are treated as predicates in the logical form representation. Their extraction process is also closely tied into the specifics of the parser, as a transformation rule is developed for each grammar rule.

This work addresses the acquisition of conceptual distinctions. In principle, it can handle any level of granularity given sufficient training data; however, addressing distinctions at the level of near-synonyms (Edmonds and Hirst, 2002) might require customized analysis for each cluster of nearly synonymous words. Inkpen and Hirst (2001) discuss how this can be automated by analyzing specialized synonymy dictionaries. Decision lists of indicative keywords are learned for the broad types of pragmatic distinctions, and these are then manually split into decision lists for more-specific distinctions.

6 Conclusion

We have presented an empirical methodology for extracting information from dictionary definitions. This differs from previous approaches by using data-driven relation disambiguation, using FrameNet se-

mantic roles annotations mapped into a reduced inventory. All the definitions from WordNet 1.7.1 were analyzed using this process, and the results evaluated by four human judges. The overall results were not high, but the evaluation was comparable to relations that were manually corrected before coding.

References

C. Barrière. 1997. *From Machine Readable Dictionaries to a Lexical Knowledge Base of Conceptual Graphs.* Ph.D. thesis, Simon Fraser University.

P. Edmonds and G. Hirst. 2002. Near-synonymy and lexical choice. *Computational Linguistics*, 28(2):105–144.

C. Fillmore, C. Wooters, and C. Baker. 2001. Building a large lexical databank which provides deep semantics. In *Proc. PACLIC-01*.

C. Fillmore. 1968. The case for case. In E. Bach and R. Harms, editors, *Universals in Linguistic Theory*. Holt, Rinehart and Winston, New York.

D. Gildea and D. Jurafsky. 2002. Automatic labeling of semantic roles. *Computational Linguistics*, 28(3):245–288.

S. Harabagiu, G. Miller, and D. Moldovan. 1999. WordNet 2—A morphologically and semantically enhanced resource. In *Proc. SIGLEX Workshop*.

D. Inkpen and G. Hirst. 2001. Building a lexical knowledge-base of near-synonym differences. In *Proc. WordNet and Other Lexical Resources*.

M. Marcus, G. Kim, M. Marcinkiewicz, R. MacIntyre, et al. 1994. The Penn Treebank: Annotating predicate argument structure. In *Proc. ARPA Human Language Technology Workshop*.

G. Miller, M. Chodorow, S. Landes, C. Leacock, and R. Thomas. 1994. Using a semantic concordance for sense identification. In *Proc. ARPA Human Language Technology Workshop*.

G. Miller. 1990. Introduction. *International Journal of Lexicography*, 3(4).

A. Novischi. 2002. Accurate semantic annotations via pattern matching. In *Proc. FLAIRS 2002*.

T. O'Hara and J. Wiebe. 2003. Preposition semantic classification via Penn Treebank and FrameNet. In *Proc. CoNLL-03*.

T. O'Hara. forthcoming. *Empirical acquisition of conceptual distinctions via dictionary definitions.* Ph.D. thesis, New Mexico State University.

P. Resnik. 1995. Disambiguating noun groupings with respect to WordNet senses. In *Proc. WVLC*.

V. Rus. 2002. *Logic Forms for WordNet Glosses.* Ph.D. thesis, Southern Methodist University.

D. Sleator and D. Temperley. 1993. Parsing English with a link grammar. In *Proc. Workshop on Parsing Technologies*.

E. Smith and D. Medin. 1981. *Categories and Concepts.* Harvard University Press, Cambridge, MA.

L. Vanderwende. 1996. *Understanding Noun Compounds using Semantic Information Extracted from On-Line Dictionaries.* Ph.D. thesis, Georgetown University.