ARTICLE IN PRESS

Available online at www.sciencedirect.com

SCIENCE DIRECT®

ELSEVIER

Computer Speech and Language 18 (2004) 297–313

COMPUTER SPEECH AND LANGUAGE

www.elsevier.com/locate/csl

Word sense disambiguation of WordNet glosses

Dan Moldovan *, Adrian Novischi

Department of Computer Science, University of Texas at Dallas, Richardson, TX 75083-0688, USA

Received 2 October 2003; received in revised form 11 May 2004; accepted 20 May 2004

Abstract

This paper presents a suite of methods and results for the semantic disambiguation of WordNet glosses. WordNet is a resource widely used in natural language processing and artificial intelligence. Intended and designed as a lexical database, WordNet exhibits some deficiencies when used as a knowledge base. By semantically disambiguating the words in the glosses, we add pointers from each word to its concept or synset, and this increases the connectivity between the WordNet concepts by approximately an order of magnitude. We show how lexical chains and other applications can be built on this richly connected WordNet. The semantic disambiguation of the WordNet glosses is performed using automatic methods based on a set of heuristics. The precision of the semantic annotation is improved by using voting between the disambiguation system described here and another WSD system. The entire WordNet 2.0 has been disambiguated with an overall precision of 86% and is available at http://xwn.hlt.utdallas.edu.

1. Introduction

Many artificial intelligence problems, like natural language understanding, require extensive common sense knowledge. Over the years, there were many attempts to overcome the lack of publicly available large knowledge bases by using WordNet, although the initial intent of this electronic lexical database was not to serve as a knowledge base (Miller, 1995). The entries in WordNet are sets of synonyms, called synsets, each representing a concept. Each synset has associated with it a gloss that contains one or more definitions and some examples. Between the concepts there are semantic relations, the most common relation between nouns is the ISA relation (or HYPERNYMY) that organizes the noun synsets into hierarchies. Verbs are grouped in

* Corresponding author.
E-mail addresses: moldovan@utdallas.edu (D. Moldovan), adrian.novischi@student.utdallas.edu (A. Novischi).

0885-2308/$ - see front matter
doi:10.1016/j.csl.2004.05.007

clusters corresponding to semantic domains including: verbs of change, communication, cognition, creation, emotion, etc. Verb synsets are also organized into hierarchies by the HYPERNYMY relation and between verb synsets there are various relations like TROPONOMY, ENTAIL and CAUSE. The adjectives and adverbs are organized in pairs of clusters of opposite meanings. In designing WordNet an effort was made to keep the set of semantic relations small, to be generally applicable, and to provide some connectivity between concepts.

1.1. The problem of word sense disambiguation

Polysemy, or lexical ambiguity, is the property of some words to have multiple meanings or senses. The definition and the number of senses is a function of the dictionary used. In our case, naturally, we use the senses of the words as defined in WordNet. In general, the word sense disambiguation (WSD) problem is the labeling of each content word with the most appropriate sense. A polysemous word alone cannot be disambiguated since there is no context to discriminate between its senses; one or more surrounding words are necessary to identify the meaning of a word. Sometimes only one adjacent word is sufficient to define the sense of a word, other times an entire sentence or paragraph is necessary to provide enough context to determine the sense of a word correctly.

1.2. WordNet limitations

WordNet was designed more as a dictionary based on psycho-linguistic principles than as a knowledge base. From this point of view WordNet has only a limited number of connections between topically related words (Harabagiu and Moldovan, 1998). For example in WordNet 2.0 there is no link between the verb *eat* with sense 1:

"eat#1 – take in solid food"

and the noun *refrigerator* with sense 1:

"refrigerator#1 – home appliance in which food can be stored at low temperatures"

One would have to detect that concept *food #1* is common to both glosses, and semantically relates the two concepts.

In this paper, we attempt to overcome the low WordNet connectivity by semantically disambiguating all open class words (i.e., nouns, verbs, adjectives and adverbs) in the glosses. This way, each word in a gloss is linked with its corresponding concept. By adding links between the synset of the gloss and the disambiguated words in the gloss (Harabagiu and Moldovan, 1998), one can increase the connectivity of WordNet concepts by approximately an order of magnitude. This, in turn, facilitates the construction of lexical chains between concepts which is useful in discovering text inferences, i.e., new information which is not stated explicitly in a text (Harabagiu and Moldovan, 1998).

1.3. Extended WordNet

The word sense disambiguation of the WordNet glosses is part of a larger project, called eXtended WordNet (XWN), (http://xwn.hlt.utdallas.edu), in which the information contained in the glosses is brought to bear. The process of extending WordNet with these new features consists of four tasks:

(1) *Preprocessing*. This stage involves the separation of the glosses into definitions and examples, tokenization, identification of compound words, and part of speech tagging.
(2) *Word sense disambiguation*. All open class words in the glosses are automatically tagged with the appropriate WordNet senses. The words are linked to their corresponding synsets.
(3) *Logic form transformations*. The glosses are parsed and transformed into logic forms enabling applications such as text inference or axiomatic logic proofs.
(4) *Topical relations*. A larger set of connections can be established among words, independent of their part of speech, based on their association with a particular context or topic. Such connections, called topical relations, are useful in information retrieval, information extraction, text coherence and other applications.

In this paper, we are concerned with the semantic disambiguation of all open class words in WordNet glosses.

1.4. Related work

The desire to build knowledge bases for common sense reasoning motivated the efforts to extract information from dictionary definitions. Semantic relations between words in dictionary definitions were detected using pattern matching (Chodorow, 1985; Agirre et al., 1994; Alshawi, 1987). The extraction of semantic relations from dictionaries was based on parsing the definitions using specially designed or general-purpose syntactic parsers (Wilks et al., 1996).

Work related to the automatic construction of lexical knowledge bases from machine readable dictionaries is presented in William et al. (1998).

There is considerable amount of work in the WSD research area and the studies differ with the sense inventory, the types of text involved and the target words (Ide and Veronis, 1998). The WSD methods can be broadly classified in:

(1) data-driven methods based on statistics on large corpora containing instances of target words, and
(2) knowledge-driven methods that use external knowledge sources for the semantic disambiguation.

1.5. What is specific to WSD of WordNet glosses?

The semantic disambiguation of WordNet glosses is different from the semantic disambiguation of open text. First, we know the concept to which the gloss belongs. The concept influences the senses of the words in that gloss. This is valuable information that can be exploited.

Second, the glosses are different than open text. They do not have the structure of a sentence; the glosses represent grammatically incomplete sentences. The absence of some words makes the glosses' disambiguation task harder than the disambiguation of open text. For example, the gloss of *second* #5 is *following the first in an ordering or series*. The disambiguation of the word *following* is more difficult than it would be in a sentence like *"Second means following the first..."*.

Third, glosses have an idiosyncratic nature, containing patterns in their definitions. Usually the open class words contained in these patterns have the same sense in all occurrences, facilitating the disambiguation of many such words.

Fourth, since each synset is an entry in WordNet, we cannot easily use the "one sense per discourse" idea observed by Yarowsky (Yarowsky, 1995).

Statistical methods cannot be used to train a machine learning algorithm on a set of manually disambiguated glosses because: (1) statistical methods work well only when applied to a few words for which there are sufficient training data, while our goal is to disambiguate all open class words and (2) a lot of words appear only a few times in WordNet glosses and do not exhibit the entire range of their senses; thus it is not possible to develop a sufficiently large training set of examples in many cases.

Because it is difficult to obtain a semantically tagged corpus and the features appearing in such corpus are very sparse, machine learning techniques were not found to be very successful. Rather, our experiments show that a series of methods based on heuristics is a more suitable approach for the semantic disambiguation of the WordNet glosses.

The methods used for the semantic disambiguation are discussed in Section 2 while results are presents in Section 3. The main sources of errors are analyzed in Section 4, and an application of the WordNet disambiguated glosses is presented in Section 5. Finally, conclusions are drawn in Section 6.

2. Semantic disambiguation methods

As explained in the introduction, statistical methods cannot be trained successfully on the sets of manually disambiguated glosses because of the sparse occurrence of words and their senses. Manual disambiguation is known to be very laborious and time intensive thus, it is rather difficult to obtain an adequate corpus for training. However, a set of manually disambiguated glosses can be used for testing the disambiguation algorithm. To overcome the data sparsity problem we rely on a set of methods that disambiguate classes of words that share a common property. These methods are used collectively to complete the job. We also rely on external resources like SemCor (Miller et al., 1994) or domain labels of glosses (Magnini and Cavaglia, 2000).

SemCor is a corpus consisting of about 25% of the Brown corpus files having all the words tagged with their part of speech, and the content words semantically disambiguated. The SemCor corpus was annotated with senses from WordNet 1.6. In order to make it useful for the disambiguation of WordNet 2.0 glosses, we automatically transformed the senses from SemCor WordNet 1.6 to WordNet 2.0. Using the domain labels assigned to noun synsets in WordNet 1.6, we automatically assigned a domain label to 65,913 noun synsets in WordNet 2.0 using a mapping between the two WordNet versions.

Before disambiguating the glosses, they are preprocessed, separating the glosses into definitions and examples, performing part of speech tagging, and identifying compound concepts.

Some methods used for the disambiguation were first introduced in Harabagiu et al. (1999). The use of the repetitive patterns in the WordNet glosses was first described in Novischi (2002). The rest of the methods are presented here for the first time.

2.1. Monosemous words

This method identifies all the words with only one sense and marks them with sense #1.

Example. Sense 2 of the noun *masterpiece* has the gloss:
"masterpiece#2 – an outstanding achievement"
The noun *achievement* has only one sense in WordNet and will be marked with sense #1.

2.2. Same hierarchy relation

This method assigns, to a noun or verb in a gloss, the sense that is an ancestor of the synset of the gloss. The rationale of this heuristic is that a concept is defined by specifying its general class and then adding the characteristic features.

Example 1. The gloss of the noun *relic* with sense 1 is:
"relic#1 – an antiquity that has survived from the distant past"

Sense 3 of the noun *antiquity* is a direct hypernym of the concept *relic#1* and this is the sense selected by the method.

Example 2. The gloss of the verb concept *caramelize#1* is:
"caramelize#1 – be converted into caramel"
Sense 8 of the verb *convert* is a direct hypernym of *caramelize#1* so we tag it accordingly.

2.3. Lexical parallelism

This method identifies the words with the same part of speech, separated by commas or conjunctions, and marks them with the senses that belong to the same hierarchy (for nouns and verbs), or to the same cluster (for adjectives). This method relies on the intuition that lexically parallel words are semantically related.

Example. The gloss of the noun concept *game*#3 is:
"game#3 – an amusement or pastime"

Sense 2 of the noun *amusement* and sense 1 of the noun *pastime* have the same hypernym *diversion#1*, *recreation#1* so both nouns are disambiguated.

2.4. SemCor bigrams

Given a word in a gloss, the SemCor bigrams method forms two pairs; one pair contains the current word lemma together with the previous word lemma, and the second pair contains the current word lemma and the next word lemma, and searches for these pairs in the SemCor corpus. If in all occurrences, of either of these pairs, the given word has the same sense, and the number of occurrences is bigger than a threshold, then we assign that sense to the word. When forming the pairs, we do not consider determiners or conjunctions since these types of words do not give any clue regarding the sense of the target word.

Example 1. The gloss of sense 1 of the noun *remake* is
"creation that is created again or anew".

For the verb *created*, we form a pair using its lemma *create* and the lemma of the previous word *be*, and search the SemCor corpus for all occurrences of this pair. We find 4 occurrences in which the verb *create* has sense 2:
"Existence is created#2 and willed by God"
"He was created#2 not exactly immortal"
"Man was created#2 with capacity of immortality"
"Adam was created#2 with the capacity of growing old"
Therefore, the SemCor bigrams method tags the verb *created* with sense 2.

Example 2. In the gloss of the concept *screen door*#1, *screen*#7:
"screen_door#1 screen#7 – a door that is a screen to keep insects from entering a building through the open door"

we take the adjective *open* and form a pair with the next word the noun *door*. We search the phrase *open door* in SemCor corpus and we find the following occurrences:
"the black line of the open#1 door."
"Juanita stopped just behind the open#1 door."
In these two occurrences the adjective *open* was tagged with the sense 1. Therefore, in the gloss above the adjective *door is* tagged with sense 1. In the same way, making a pair with the previous word *open*, the noun *door* is also disambiguated with sense 1.

2.5. Cross-reference

Given an ambiguous word *w* in synset *S*, the cross-reference method looks for a reference to the synset in all the glosses corresponding to word *w* senses. By reference to a word *w* we understand a word, or a part of a compound concept, that has the same lemma as the word *w*.

The rationale of this heuristic is the fact that if two synsets contain references to each other they must be semantically related.

Example. In the gloss of the noun concept *silver_screen #2*:
"screen#1, silver screen #2, projection screen #1 – a white or silvered surface where pictures can be projected for viewing"
the verb *project* has 12 senses. Only sense 4 of this verb contains the word *screen* in its gloss:
"project#4 – project on a screen"
therefore we mark the verb *project* with sense 4.

2.6. Reversed cross-reference

Given a word *w* in the gloss *G* of the synset *S*, the reversed cross-reference method assigns to the word *w* the sense that contains in its set of synonyms one of the words from the gloss *G*.

The rationale of this method is that, if two word forms appear in the same set of synonyms and also in the same gloss, they must be semantically related.

Example. The gloss

"rifle ball#1 – a bullet designed to be fired from a rifle; no longer made spherical in shape"

contains the noun *shape*. The *adjective* spherical has 2 senses. The set of synonyms in sense 2 of this adjective contains the word *shaped*: *ball-shaped*, *global*, *globose*, *globular*, *orbicular*, *spheric*, *spherical*. The word *shaped* from the set of synonyms and the noun *shape* from the gloss have the same stem, therefore we assign sense 2 to the adjective *shape*.

2.7. Distance among glosses

This method determines the number of common word lemmas between two synsets. For an ambiguous word w in a gloss G, this method selects the sense of the word that has in its gloss the greatest number of common word lemmas with the gloss G.

The rationale of this method is to select the sense of a word that is most semantically related with the gloss G. Therefore, we select the sense that has the greatest number of common words with the gloss G. This algorithm was first proposed by Lesk (1986).

Example. Sense 2 of the noun *screening* has the following gloss:

"screen#4 – fabric of metal or plastic mesh".

Sense 4 of the noun *mesh* is the only one that contains the noun *fabric* in its gloss:

"net#6, network#3, mesh#4, meshing#2, meshwork#1 – an open fabric of string or rope or wire woven together at regular intervals".

Both glosses have the noun *fabric* in common, therefore we tag the noun *mesh* with sense 4.

2.8. Common domain

Using domain synset labels, the Common Domain method selects the sense of a word that has the same domain label as the synset of the gloss.

The rationale behind this method is that the senses of the words used in the gloss G of the synset S must have the same domain as the synset S.

Example. The following gloss:

"regulator#1 – any of various controls or devices for regulating or controlling fluid flow, pressure, temperature, etc."

is under the *mechanics* domain. Sense 10 of the noun *controls* is the only one that has the same domain *mechanics* as the synset of the gloss, therefore we tag the noun *controls* with sense 10.

"control#10, controller#3 – a mechanism that controls the operation of a machine".

2.9. Patterns

This method (Novischi, 2002) exploits the idiosyncratic nature of the WordNet glosses identifying repetitive expressions. To find these repetitive expressions we look for patterns of the form <*N successive words*> and <*M words… N words*>. The patterns contain the word forms and their part of speech as given by the Brill tagger, (Brill, 1995). To avoid extracting meaningless patterns when counting the number of words, we do not consider punctuation or words with the following part of speech: CC, DT, FW, SYM, LS, UH. We select only the patterns that contain at least one open class word. These patterns were sorted in decreasing order of the number of occurrences in WordNet glosses. From these patterns we selected those that contain words that have the same sense in all occurrences. The patterns were augmented with the senses of the open class words and further used to disambiguate the WordNet glosses.

Example. Given the following set of glosses:

"genus Lepidobotrys#1 – a genus of dicotyledonous trees belonging to the family Lepidobotryaceae"

"Cynoglossum#1, genus Cynoglossum#1 – a large genus of tall rough herbs belonging to the family Boraginaceae"

"Acalypha#1, genus Acalypha#1 – a genus of herbs and shrubs belonging to the family Euphorbiaceae".

We observe the pattern <*genus… belonging to the family*> occurring in all three glosses. In this pattern the words *genus*, *family* and *belong* have the same sense:

"genus#2 – (biology) taxonomic group containing one or more species"

"family#6 – (biology) a taxonomic group containing one or more genera"

"belong#5 – be classified with"

This pattern occurs 40 times in WordNet glosses. The words in this and other such patterns can be disambiguated manually once, then used automatically to disambiguate words whenever patterns occur.

2.10. First sense restricted

One commonly used baseline method for Word Sense Disambiguation is to tag the words with their sense 1 from WordNet. In order to improve this method we can ask the following questions: "When is the tagging with sense 1 wrong?" or "When is the tagging with sense 1 good?" By manually tagging sets of WordNet glosses, we observed that since the open class words are used to define concepts, they are usually used with their most general sense and are not related to a specific topic (the words that are specific to a particular topic are disambiguated by the "common domains" method). We observed that a sense of a noun or verb is more general if it has the smallest number of ancestors from all senses in the ISA hierarchy. Also a sense of an adjective is more general if it has the largest number of similarity pointers from all senses. We observed that these conditions tend to be true for sense 1 in WordNet but not for the other senses. Therefore, we built a method that selects sense 1 of a noun or verb only if this sense has the smallest number of ancestors in the ISA hierarchy, and select sense 1 of an adjective only if it has the largest number of similarity pointers of all the other senses. Although this method has limitations, its precision is above the baseline and its coverage is large.

2.11. Building the WSD system using the methods

Using all the methods presented above, we built a program, called XWN_WSD, for disambiguating the WordNet glosses. Some of the words are disambiguated by several methods. Between two different senses given by two methods, we choose the sense given by the most accurate method. To determine the accuracy of each method we tested them individually and collectively on a gold standard consisting of 3196 manually disambiguated glosses containing a total of 7977 open class words. The results are shown in Table 1. The methods are listed in the decreasing order of their precision. We built the XWN_WSD program by calling the methods in this order.

3. Results

3.1. Contribution of each method

Table 1 presents the coverage and precision results obtained by applying each method, one at a time, to a set of 3196 glosses, as well as the coverage and precision for all methods combined. The *coverage* is defined as the ratio between the number of attempted words and the total number of words. The *precision* is defined as the ratio between the correctly disambiguated words and the number of words attempted. The monosemous words were not included in the experiments reported in Tables 1 and 2.

One interesting aspect is the contribution of each method to the overall performance. To determine this, we ran the system containing all but each method at a time in order to see the penalty for not using that method. From Table 2 we see that the methods that contribute most are *Same Hierarchy* that adds a 5% precision, and *First Sense Restricted that* adds 10% coverage but decreases precision with about 2%.

Table 1
Precision and coverage for WSD methods

Method	Correct	Attempted	Coverage (%)	Precision (%)
Base line (first sense)	2087	3958	100.00	52.58
Same hierarchy	941	996	25.16	94.48
Patterns	329	398	9.90	82.66
Domains	850	1042	26.32	81.57
Lexical parallelism	502	687	17.36	73.07
Gloss distance	479	755	19.07	63.44
SemCor digrams	337	540	13.64	62.37
Cross-reference	159	255	6.44	62.35
First sense restricted	1121	1935	48.88	57.93
Reversed cross-reference	113	198	5.50	57.07
All methods	2619	3446	67.16	76.00

Table 2
Penalty in precision and coverage for each WSD method

Method	Penalty for not using			
	Correct	Attempted	Coverage (%)	Precision (%)
Same hierarchy	−283	−149	−3.38	−5.14
Patterns	−71	−44	−1.06	−1.10
Domains	−163	−102	−2.58	−2.56
Lexical parallelism	−109	−111	−2.80	−0.51
SemCor digrams	−62	−86	−2.17	+0.10
Gloss distance	−62	−115	−2.90	+0.76
Cross-reference	−16	−16	−0.40	−0.07
Reversed cross-reference	−4	−13	−0.32	+0.17
First sense restricted	−413	−416	−10.51	+1.84

3.2. Voting

Our goal is to disambiguate the open class words from WordNet glosses with high precision. Thus, in the trade-off precision versus coverage, we prefer to have a smaller set of disambiguated words with as high precision as possible. The precision of the semantic disambiguation can be improved by using voting between two disambiguation programs. Together with our XWN_WSD program, we used an existing in-house system for disambiguating words in open text built on slightly different principles. We participated with this system in Senseval 3. That system is a combination of methods consisting of hand-coded rules, contextual clues extracted from SemCor corpus, heuristics based on WordNet glosses, WordNet relations and conceptual density. To use that system, we transformed all the glosses into sentences. Then, the disambiguated sentences were transformed back into gloss forms. By comparing the output of the two WSD systems, we selected the set of words in the glosses that were assigned the same sense. We called this the "silver standard" words since, although, not checked by hand, their disambiguation was provided by the agreement between two independent systems.

Table 3 contains the results of several experiments performed on the gold-standard of 3196 glosses with 7977 content words. Line (a) contains the results obtained with the XWN_WSD system presented in this paper. Line (b) contains the results obtained with the second WSD system

Table 3
Voting between two disambiguation systems for WordNet glosses

	System	Correct	Attempted	Coverage (%)	Precision (%)
(a)	XWN_WSD	2619	3446	66.17	76.00
(b)	Open text system	965	1451	24.38	66.50
(c)	Voting	843	996	21.30	84.64
(d)	XWN_WSD + first sense	2873	3927	99.21	73.60
(e)	Open text + first sense	2127	3883	98.10	54.77
(f)	Voting + first sense	2142	3927	99.21	58.97
(g)	XWN_WSD + first sense + monosemous	6892	7946	99.61	86.73

+ *First sense* means to assign sense #1 to the words that were not disambiguated by the system.

Table 4
Precision for each part of speech checked against manually disambiguated glosses

Part of speech	Total	Mono	Poly	Correct	Attempted	Coverage (%)	Precision (%)	Baseline (%)
Noun	6176	3458	2178	2124	2705	99.52	80.09	50.09
Verb	911	129	782	438	764	97.69	57.32	52.36
Adjective	806	390	416	280	416	100.00	67.31	67.55
Adverb	84	42	42	31	42	100.00	73.81	69.05
All	7977	4019	3958	2873	3927	99.21	73.60	52.58

This experiment refers to line (d) line of Table 3, i.e., XWN_WSD + first sense.

mentioned above. Line (c) shows the results obtained by applying voting between these two systems. In lines (d) and (f), we show the results obtained by tagging with sense 1 the rest of the words that were not disambiguated in lines (a) and (b), respectively. The effect of this procedure is an increase of coverage to almost 100% at the expense of precision drop. Line (f) shows the results obtained with voting between systems (d) and (e). Finally, in line (g) we introduce for the first time the effect of the monosemous words, which does not affect the coverage but increases the precision to 86%. This is the result a user sees when using the system.

We wanted to know the system performance for different parts of speech. For this we used system (d), i.e., first sense included, but without monosemous words. The results are presented in Table 4. The system seems to best disambiguate the nouns. The hardest to disambiguate are the verbs for which the system gets only 57% in precision.

4. Error analysis

Although the methods used for the disambiguation are correct in many instances, there are cases when they incorrectly select the sense of the word. To learn more about these heuristics, we produced a detailed error analysis for each method. From the many error examples inspected, conclusions can be drawn regarding the limitations of each method.

4.1. Same hierarchy relation

This method incorrectly selects the sense of the noun that is not the head of the gloss. For example, the gloss of the synset *"reservoir#2*, *artificial lake#1"* is:

"reservoir#2, artificial lake#1 – lake used to store water for community use".

Sense 2 of the noun *water*, which has the meaning related to body of water, is a hypernym of *reservoir#2*, although the noun *water* in this gloss has sense 1, having the meaning of *liquid.*

4.2. Lexical parallelism

This method translates the lexical parallelism of the words into a semantic parallelism thus imposing the condition that lexically parallel nouns or verbs be in the same hierarchy. Although this condition holds almost all the time there are cases when it does not, e.g.:

1. Multiple pairs of senses from different hierarchies.

The currently implemented method shows only the first pair, but other pairs of senses are available.

Example. In the gloss of the concept *cheat#2*:

"cheat#2, chouse#1, shaft#2, screw#5, chicane #1, jockey#1 – defeat someone in an expectation through trickery or deceit"

sense 1 of the noun *trickery* and sense 2 of the noun *deceit* have the same hypernym: *statement#1*. Also sense 2 of the noun *trickery* and sense 3 of the noun *deceit* have the same hypernym *"act#2"*.

2. Too general hypernym.

Some words, although lexically parallel, do not have a hypernym that contains the shared semantic of the two concepts. The hypernym that is common between the two is too general for both concepts and often leads to a wrong choice of senses.

Example 1. Sense 59 of the verb *break* has the gloss:

"break#59 – weaken or destroy in spirit or body".

Sense 3 of the noun *spirit* and sense 8 of the noun *body* have the common hypernym which is the concept *attribute#2*. But this hypernym is too general and does not contain the shared semantic characteristic: both concepts are related to humans. Although sense 3 for the noun *spirit* is correct, the right sense for *body* is 1 instead of 8.

Example 2. The gloss for the verb *"take_up"* is *"take up time or space"*. Senses 1, 2, 4, 5, 6, 7 of the noun *space* and senses 2, 3, 4, 5, 7, 9, 10 of the noun *time* have as ancestor the noun concept *abstraction#6*. This common ancestor does not help to disambiguate neither the noun *time* nor the noun *space.*

3. Words that are not lexically parallel.

There are words linked by conjunctions that are not lexically parallel although they have the same part of speech. For example, in the gloss of the verb concept *let#1*:

"let#1, allow#1, permit#2 – make it possible through a specific action or lack of action for something to happen"

we have the expression *action or lack of action* in which the nouns *action* and *lack* are not linguistically parallel. Still, sense 2 of the noun *action* and sense 1 of the noun *lack* have the same hypernym: the concept *state#4*. This method wrongly disambiguates the noun *action* which has sense 4.

4.3. SemCor bigrams

The SemCor bigrams method incorrectly selects the sense of a word in the following situations:

1. Some words in front of the target word do not help to disambiguate it. For example, in the following gloss:

"look_out#2 – to protect someone's interests"

the possessive "'s" does not help to disambiguate the noun *interest*.

2. Some words following the target word do not help to disambiguate it. In the gloss of verb *display#3*:

"display #3 – attract attention by displaying some body part or posing; of animals"

the noun *attention* has sense 3 (general interest), however, in SemCor we only find the example:

I am not aware of great attention#2 by any of these authors.

in which the noun *attention* was tagged with sense 2. The preposition *by* after the noun *attention* does not help to disambiguate it.

4.4. Cross-reference

Examples of errors made by the cross-reference method are:

1. Circular reference.

In the following gloss:

"connect#3, link#3, link up#1, join#5, unite#4 – be or become joined or united or linked"

the human annotator tagged the verb *join* with sense 2, verb *unite* with sense 5 and verb *link* with sense 2:

"join#2, bring together#1 – cause to become joined or linked"

"unite#5, unify #4, merge#3 – join or combine"

"connect #1, link #2, tie#4, link up#2 – connect, fasten, or put together two or more pieces".

But this gloss belongs to the synset that is sense 5 for the verb *join*, sense 4 for the verb *unite*, and sense 3 for the verb *link*, and these are the senses selected by the cross-reference method.

2. Words with the same stem, but with a different part of speech and unrelated meanings.

In the gloss:

"submit#2, state#2, put forward#1, posit#2 – put before"

sense 2 of the verb *put* has the gloss:

"put#2 – cause to be in a certain state; cause to be in a certain relation"

This gloss contains the noun *state*, although in the set of synonyms of the original synset we have the verb *state*. The noun *state* (sense 4) referring to the main attributes of an entity is not related to the verb *state* (sense 2) having the meaning of advise, suggest or propose.

3. Words that are part of a compound concept with a different meaning.

Given a word in the gloss of synset *S*, the cross-reference method assigns the sense that has a reference to one of the synonyms of the synset *S*. But sometimes the reference is only a part of a compound concept that does not have any connection with the original word meaning. For example, given the gloss of sense 2 of the verb *make up*

"make up#2 – devise or compose"

the human annotator assigned for the verb *compose* sense 4:

"compose#4, compile#2 – put together out of existing material"

However, the cross-reference method wrongly assigned sense 5:

"compose#5 – (calm someone, especially oneself); make quiet"

because the gloss contains the word *make* which is a part of the verb *make up.*

4.5. Reversed cross-reference

An example of error made by this method is:

1. Reference with the same stem but unrelated meanings

When trying to find the sense of a target word w in a gloss G we find another word w_i that has the same stem with a word w_j in the set of synonyms, but the meanings of words w_i and w_j are not related.

Example. Given the gloss:

"fixate#3,t fix#8 – make fixed, stable or stationary"

the correct sense for the verb *make* is sense 2:

"make#2, get#5 – give certain properties to something".

However, the reversed cross-reference method found sense 28 that contains the verb *fix* in the set of synonyms which has the same stem as the adjective *fixed* from the gloss. In the gloss of the synset *fixate*, *fix* the adjective *fixed* means *not moving* while the verb *fix* in the set of synonyms of sense 28 of the verb *make* has the sense of *create* or *prepare*.

4.6. Distance among glosses

The errors introduced by this method are caused by:

1. Noise words.

Sometimes the common words (between gloss G that is disambiguated and the gloss of a wrong sense of a word in gloss G) are not related to the meaning of the synset; these words are just used to build the definition.

Example. Given the following gloss:

"conduct#4, transmit#2, convey#5, carry #3, channel#1 – transmit or serve as the medium for transmission"

we try to disambiguate the noun *medium*. Sense 9 of this noun contains also the verb *serve*:

"medium, spiritualist – someone who serves as an intermediary between the living and the dead"

However, the correct sense of the noun *medium* is sense 1. The verb *serve*, although appears in both glosses, is not related to either of them. This verb is used only for building the definitions, that could be formulated using different words. The other observation is that the verb *serve* accepts many types of arguments in the prepositional attachment starting with as:

"serve as medium of transmission"

"serve as intermediary"

4.7. Common domain

In WordNet, there are concepts that have definitions referring to the action of verbs having noun arguments that are not in the same domain. However, there are different senses of these noun arguments that are in the same domain as the concept being defined.

Example. The gloss of sense 1 of noun *emigration* is:
"emigration#1, out-migration#1, expatriation #2 – migration from a place (especially migration from your native country in order to settle in another)"
This gloss has the domain *sociology*. In this gloss, we see that the noun *place*, is an argument of the verb *migrate*, described by the noun *migration*. The correct sense of the noun *place* is sense 2:
"place#2, property#1 – any area set aside for a particular purpose"
This sense has the general domain *factotum*. However, sense 10 of noun *place* whose definition is:
"place#10, station#2 – proper or designated social situation"
has the same domain *sociology* as the noun *migration* with sense 1.

4.8. Patterns

Although we can choose to disambiguate these patterns manually, some open class words in the patterns can have different, but very close senses, or can have different interpretations.

Example. Let us consider the pattern <*"in... manner"*>. This pattern occurs 1774 times in WordNet glosses being one of the most productive patterns. It occurs especially in adverb glosses. Examples:
"sportively#1 – in a merry sportive manner"
"contradictorily#1 – in a contradictory manner"
"mercifully#2, with_mercy#1, showing_mercy#1 – in a compassionate manner"
"helpfully#1 – in a helpful manner"

In this pattern, the noun *manner* can be replaced with the noun *way*:
"sportively#1 – in a merry sportive way"
"contradictorily #1 – in a contradictory way"
"mercifully#2, with_mercy#1, showing_mercy#1 – in a compassionate way"
"helpfully#1 – in a helpful way"
Therefore, we can assign sense 1 to the noun *manner*:
"manner#1, mode#1, style#2, way#1, fashion#1 – how something is done or how it happens"
However, the glosses for the adverbs *sportively*, *mercifully* and *helpfully* also refer to human behavior, and in this case sense 2 is assigned.
"manner#2, personal manner#1 – a way of acting or behaving"

5. Applications

5.1. Lexical chains used in question answering

A major problem in Question Answering is often that an answer is expressed with words different from the question keywords. In such cases, it is useful to find topically related words to the question keywords. By exploiting the information in the WordNet glosses, the connectivity between the synsets is dramatically increased. When a word in a gloss is semantically disambiguated, it points to the synset it belongs to. In the context of XWN, or any other lexical database, topical relations can be expressed as *lexical chains*. These are sequences of semantically related words that link two concepts. Lexical chains have been used in computational linguistics to study: discourse, coherence, inference, implicatures, malapropisms and others. Lexical chains improve the performance of question answering systems in two ways: (1) increase the document retrieval recall and (2) improve the answer extraction by providing the much needed world knowledge axioms that link question keywords with answer concepts.

It is possible to establish connections between synsets via topical relations. We developed software that automatically provides connecting paths between any two WordNet synsets S_i and S_j up to a certain distance. The meaning of these paths is that the concepts along a path are topically related.

We illustrate, with several examples of questions from TREC, the usefulness of constructing lexical chains that link question keywords with answer words (Moldovan et al., 2003).

Q1403: When was the internal combustion engine invented?

Answer: The first internal combustion engine was built in 1867

Lexical chains:

(1) invent:v#1 → HYPERNYM → create_by_mental_act:v#1 → HYPERNYM → create:v#l → HYPONYM → build:v#1

Q1462: Where is the oldest synagogue in the United States?

Answer: Newport is marking the 350th anniversary of the founding of Trinity Church, and is also home to the nation's oldest synagogue

Lexical chains:

(1) United_States:n#1 → HYPERNYM → North_American_country:n#1 → HYPERNYM → country:n#1 → GLOSS → nation:n#1

Q1446: How did Mahatma Gandhi die?

Answer: After reaching an agreement with the South African government on Indian rights, Gandhi returned to India in 1914, eventually leading his country to full independence from Britain in 1947. He was shot dead by a Hindu fanatic.

Lexical Chains

(1) die:v#1 → RGLOSS → kill:v#1 → HYPONYM → shoot:v#2

5.2. General-purpose word sense disambiguation

Enriching WordNet with links between the correct senses of topically related words can improve the precision of word sense disambiguation methods in general. For example, in the simplest case, we can consider for each sense of an ambiguous word w the set of topically related nouns. Then we can compute how accurately each set matches the context of the ambiguous word in the text using a variant of the Lesk algorithm (Lesk, 1986; Kilgarriff and Rosenzweig, 2000). In addition, XWN can serve as a training corpus for other WSD systems.

6. Conclusion

A suite of heuristical methods are presented for the disambiguation of WordNet glosses. Unfortunately, due to the sparsity of training data in WordNet, learning methods cannot be used.

The methods are called in order of their known accuracy. Each method increases the pool of disambiguated words. The words left unprocessed, after all methods have been applied, are assigned the default sense 1.

Including the monosemous words, this approach provides an accuracy of 86% measured on a set of gold standard glosses disambiguated by human.

Once the WordNet glosses are disambiguated, several applications become possible. One application discussed in the paper is the utilization of lexical chains to Question Answering. We have used lexical chains successfully to link question keywords with answer text and provide axioms to a QA logic prover.

References

Agirre, E., Arregi, X., Artola, X., de Ilarraza, D., Sarasola, K., 1994. A methodology for the extraction of semantic knowledge from dictionaries using phrasal patterns.

Alshawi, H., 1987. Processing dictionary definitions with phrasal pattern hierarchies. Computational Linguistics 13 (3–4), 195–202.

Brill, E., 1995. Transformation-based error driven learning and natural language processing: a case study in part-of-speech tagging. Computational Linguistics 21 (4), 543–566.

Chodorow, M.S., 1985. Extracting semantic hierarchies from a large on-line dictionary. In: Proceedings of the 23th ACL. Chicago, IL, pp. 299–304.

Harabagiu, S., Miller, G., Moldovan, D., 1999. WordNet 2 – a morphologically and semantically enhanced resource. In: Proceedings of SIGLEX-99. University of Maryland, pp. 1–8.

Harabagiu, S., Moldovan, D., 1998. Knowledge processing on an extended WordNet. In: Fellbaum, C. (Ed.), WordNet – An Electronic Lexical Database. MIT Press, Cambridge, MA, pp. 379–406.

Ide, N., Veronis, J., 1998. Introduction to the special issue on word sense disambiguation: the state of the art. Computational Linguistics 24 (1), 1–40.

Kilgarriff, A., Rosenzweig, J., 2000. Framework and results for English senseval. Computers and the Humanities 34, 15–48.

Lesk, M., 1986. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In: Proceedings of the ACM SIG-DOC Conference, Ontario, Canada.

Magnini, B., Cavaglia, G., 2000. Integrating subject field codes into WordNet. In: Proceedings of the LREC-2000, Athens, Greece.

Miller, G., 1995. WordNet: a lexical database. Communications of the ACM 38 (11), 39–41.

Miller, G., Chodorow, M., Landes, S., Leacock, C., Thomas, R., 1994. Using a semantic concordance for sense identication. In: Proceedings of the ARPA Human Language Technology Workshop, pp. 240–243.

Moldovan, D., Clark, C., Harabagiu, S., Maiorano, S., 2003. Cogex: a logic prover for question answering. In: Proceedings of HLT-NAACL.

Novischi, A., 2002. Accurate semantic annotations via pattern matching. In: Proceedings of Florida Artificial Intelligence Research Society, Pensacola Beach, FL, USA, pp. 375–379.

Wilks, Y.A., Slator, B.M., Guthrie, L.M., 1996. Electric Words: Dictionaries, Computers, and Meanings. MIT Press, Cambridge, MA.

William B. Dolan, Stephen, D., Richardson, L.V., 1998. Mindnet: acquiring and structuring semantic information from text.

Yarowsky, D., 1995. Unsupervised word sense disambiguation rivaling supervised methods. In: Meeting of the Association for Computational Linguistics, Cambridge, MA, pp. 189–196.