Automatic Extraction of Systematic Polysemy Using Tree-cut

Noriko Tomuro
DePaul University
School of Computer Science, Telecommunications and Information Systems
243 S. Wabash Ave.
Chicago, IL 60604
tomuro@cs.depaul.edu

Abstract

This paper describes an automatic method for extracting *systematic polysemy* from a hierarchically organized semantic lexicon (WordNet). Systematic polysemy is a set of word senses that are related in systematic and predictable ways. Our method uses a modification of a tree generalization technique used in (Li and Abe, 1998), and generates a *tree-cut*, which is a list of clusters that partition a tree. We compare the systematic relations extracted by our automatic method to manually extracted WordNet *cousins*.

1 Introduction

In recent years, several on-line broad-coverage semantic lexicons became available, including LDOCE (Procter, 1978), WordNet (Miller, 1990) and HECTOR (Kilgarriff, 1998a). These lexicons have been used as a domain-independent semantic resource as well as an evaluation criteria in various Natural Language Processing (NLP) tasks, such as Information Retrieval (IR), Information Extraction (IE) and Word Sense Disambiguation (WSD).

However, those lexicons are rather complex. For instance, WordNet (version 1.6) contains a total of over 120,000 words and 170,000 word senses, which are grouped into around 100,000 *synsets* (synonym sets). In addition to the size, word entries in those lexicon are often polysemous. For instance, 20% of the words in Wordnet have more than one sense, and the average number of senses of those polysemous words is around 3. Also, the distinction between word senses tends to be ambiguous and arbitrary. For example, the following 6 senses are listed in WordNet for the noun "door":

1. **door** – a swinging or sliding barrier
2. **door** – the space in a wall
3. **door** – anything providing a means of access (or escape)
4. **door** – a swinging or sliding barrier that will close off access into a car
5. **door** – a house that is entered via a door
6. **door** – a room that is entered via a door

Because of the high degree of ambiguity, using such complex semantic lexicons brings some serious problems to the performance of NLP systems. The first, obvious problem is the computational intractability: increased processing time needed to disambiguate multiple possibilities will necessarily slow down the system. Another problem, which has been receiving attention in the past few years, is the inaccuracy: when there is more than one sense applicable in a given context, different systems (or human individuals) may select different senses as the correct sense. Indeed, recent studies in WSD show that, when sense definitions are fine-grained, similar senses become indistinguishable to human annotators and often cause disagreement on the correct tag (Ng *et al.*, 1999; Veronis, 1998; Kilgarriff, 1998b). Also in IR and IE tasks, difference in the correct sense assignment will surely degrade recall and precision of the systems. Thus, it is apparent that, in order for a lexicon to be useful as an evaluation criteria for NLP systems, it must represent word senses at the level of granularity that captures human intuition.

In Lexical Semantics, several approaches have been proposed which organize a lexicon based on *systematic polysemy*:[1] a set of word senses that are related in systematic and predictable

[1] Systematic polysemy (in the sense we use in this paper) is also referred to as *regular polysemy* (Apresjan, 1973) or *logical polysemy* (Pustejovsky, 1995).

ways (e.g. ANIMAL and MEAT meanings of the word "chicken").[2] In particular, (Buitelaar, 1997, 1998) identified systematic relations that exist between abstract semantic concepts in the WordNet noun hierarchy, and defined a set of *underspecified semantic classes* that represent the relations. Then he extracted all polysemous nouns in WordNet according to those underspecified classes and built a lexicon called CORELEX. For example, a CORELEX class AQU (which represents a relation between ARTIFACT and QUANTITY) contains words such as "bottle", "bucket" and "spoon".

Using the abstract semantic classes and organizing a lexicon based on systematic polysemy addresses the two problems mentioned above in the following ways. For the first problem, using the abstract classes can reduce the size of the lexicon by combining several related senses into one sense; thus computation becomes more efficient. For the second problem, systematic polysemy does reflect our general intuitions on word meanings. Although the distinction between systematic vs. non-systematic relations (or related vs. unrelated meanings) is sometimes unclear, systematicity of the related senses among words is quite intuitive and has been well studied in Lexical Semantics (for example, (Apresjan, 1973; Cruse, 1986; Nunberg, 1995; Copestake and Briscoe, 1995)).

However, there is one critical issue still to be addressed: the level of granularity at which the abstract classes are defined. The problem is that, when the granularity of the abstract classes is too coarse, systematic relations defined at that level may not hold uniformly at more fine-grained levels (Vossen *et al.*, 1999). For instance, the CORELEX class AQU mentioned above also contains a word "dose".[3] Here, the relation between the senses of "dose" is different from that of "bottle", "bucket" and "spoon", which can be labeled as CONTAINER-CONTAINERFUL relation. We argue that human intuitions can distinguish meanings

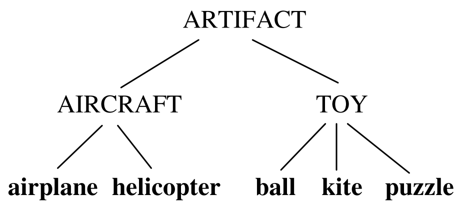

Figure 1: An example thesaurus tree

at this level, where differences between the systematic relations are rather clear, and therefore lexicons that encode word senses at this level of granularity have advantages over fine-grained as well as coarse-grained lexicons in various NLP tasks.

Another issue we like to address is the ways for extracting systematic polysemy. Most often, this procedure is done manually. For example, the current version of WordNet (1.6) encodes the similarity between word senses (or synsets) by a relation called *cousin*. But those cousin relations were identified manually by the WordNet lexicographers. A similar effort was also made in the EuroWordnet project (Vossen *et al.*, 1999). However, manually inspecting a large, complex lexicon is very time-consuming and often prone to inconsistencies.

In this paper, we propose a method which automatically extracts systematic polysemy from a hierarchically organized semantic lexicon (WordNet). Our method uses a modification of a tree generalization technique used in (Li and Abe, 1998), and generates a *tree-cut*, which is a list of clusters that partition a tree. Then, we compare the systematic relations extracted by our automatic method to the WordNet cousins. Preliminary results show that our method discovered most of the WordNet cousins as well as some more interesting relations.

2 Tree Generalization using Tree-cut and MDL

Before we present our method, we first give a brief summary of the tree-cut technique which we adopted from (Li and Abe, 1998). This technique is used to acquire generalized case frame patterns from a corpus using a thesaurus tree.

2.1 Tree-cut Models

A *thesaurus tree* is a hierarchically organized lexicon where leaf nodes encode lexical data

[2]Note that systematic polysemy should be contrasted with *homonymy* which refers to words which have more than one unrelated sense (e.g. FINANCIAL_INSTITUTION and SLOPING_LAND meanings of the word "bank").

[3]Senses of "dose" in WordNet are: (1) a measured portion of medicine taken at any one time, and (2) the quantity of an active agent (substance or radiation) taken in or absorbed at any one time.

(i.e., words) and internal nodes represent abstract semantic classes. A *tree-cut* is a partition of a thesaurus tree. It is a list of internal/leaf nodes in the tree, and each node represents a set of all leaf nodes in a subtree rooted by the node. Such set is also considered as a *cluster*.[4] Clusters in a tree-cut exhaustively cover all leaf nodes of the tree, and they are mutually disjoint. For example, for a thesaurus tree in Figure 1, there are 5 tree-cuts: [airplane, helicopter, ball, kite, puzzle], [AIRCRAFT, ball, kite, puzzle], [airplane, helicopter, TOY], [AIRCRAFT, TOY] and [ARTIFACT]. Thus, a tree-cut corresponds to one of the levels of abstraction in the tree.

Using a thesaurus tree and the idea of tree-cut, the problem of acquiring generalized case frame patters (for a fixed verb) from a corpus is to select the best tree-cut that accounts for both observed and unobserved case frame instances. In (Li and Abe, 1998), this generalization problem is viewed as a problem of selecting the best model for a tree-cut that estimates the true probability distribution, given a sample corpus data.

Formally, a *tree-cut model* M is a pair consisting of a tree-cut Γ and a probability parameter vector Θ of the same length,

$$M = (\Gamma, \Theta) \tag{1}$$

where Γ and Θ are:

$$\Gamma = [C_1, .., C_k], \Theta = [P(C_1), .., P(C_k)] \tag{2}$$

where C_i $(1 \leq i \leq k)$ is a cluster in the tree-cut, $P(C_i)$ is the probability of a cluster C_i, and $\sum_{i=1}^{k} P(C_i) = 1$. For example, suppose a corpus contained 10 instances of verb-object relation for the verb "fly", and the frequency of object noun n, denoted $f(n)$, are as follows: $f(airplane) = 5, f(helicopter) = 3, f(ball) = 0, f(kite) = 2, f(puzzle) = 0$. Then, the set of tree-cut models for the thesaurus tree shown in Figure 1 includes ([airplane, helicopter, TOY], [0.5, 0.3, 0.2]) and ([AIRCRAFT, TOY], [0.8, 0.2]).

Note that $P(C)$ is the probability of cluster $C = \{n_1, .., n_m\}$ as a whole. It is essentially the sum of all (true) probabilities of the member words, that is, $P(C) = \sum_{j=1}^{m} P(n_j)$. Here, compared to knowing all $P(n_j)$ (where $1 \leq j \leq m$) individually, knowing one $P(C)$ can only facilitate an estimate of uniform probability distribution among members as the best guess, that is, $P(n_j) = \frac{P(C)}{m}$ for all j. Therefore, in general, when clusters $C_1..C_m$ are merged and generalized to C according to the thesaurus tree, the estimation of a probability model becomes less accurate.

2.2 The MDL Principle

To select the best tree-cut model, (Li and Abe, 1998) uses the *Minimal Description Length (MDL)* principle (Rissanen, 1978). The MDL is a principle of data compression in Information Theory which states that, for a given dataset, the best model is the one which requires the minimum length (often measured in bits) to encode the model (the *model description length*) and the data (the *data description length*). For the problem of case frame generalization, the MDL principle fits very well in that it captures the trade-off between the simplicity of a model, which is measured by the number of clusters in a tree-cut, and the goodness of fit to the data, which is measured by the estimation accuracy of the probability distribution.

The calculation of the description length for a tree-cut model is as follows. Given a thesaurus tree T and a sample S consisting of the case frame instances, the total description length $L(M, S)$ for a tree-cut model $M = (\Gamma, \Theta)$ is

$$L(M, S) = L(\Gamma) + L(\Theta|\Gamma) + L(S|\Gamma, \Theta) \tag{3}$$

where $L(\Gamma)$ is the model description length, $L(\Theta|\Gamma)$ is the parameter description length (explained shortly), and $L(S|\Gamma, \Theta)$ is the data description length. Note that $L(\Gamma) + L(\Theta|\Gamma)$ essentially corresponds to the usual notion of the model description length.

Each length in $L(M, S)$ is calculated as follows.[5] The model description length $L(\Gamma)$ is

$$L(\Gamma) = log_2|G| \tag{4}$$

where G is the set of all cuts in T, and $|G|$ denotes the size of G. This value is a constant for

[4] A leaf node is also a cluster whose cardinality is 1.

[5] For justification and detailed explanation of these formulas, see (Li and Abe, 1998).

all models, thus it is omitted in the calculation of the total length.

The parameter description length $L(\Theta|\Gamma)$ indicates the complexity of the model. It is the length required to encode the probability distribution of the clusters in the tree-cut Γ. It is calculated as

$$L(\Theta|\Gamma) = \frac{k}{2} \times log_2|S| \quad (5)$$

where k is the length of Θ, and $|S|$ is the size of S.

Finally, the data description length $L(S|\Gamma,\Theta)$ is the length required to encode the whole sample data. It is calculated as

$$L(S|\Gamma,\Theta) = -\sum_{n \in S} log_2 P(n) \quad (6)$$

where, for each $n \in C$ and each $C \in \Gamma$,

$$P(n) = \frac{P(C)}{|C|} \quad (7)$$

and

$$P(C) = \frac{f(C)}{|S|} \quad (8)$$

Note here that, in (7), the probability of C is divided evenly among all n in C. This way, words that are not observed in the sample receive a non-zero probability, and the data sparseness problem is avoided.

Then, the best model is the one which requires the minimum total description length.

Figure 2 shows the MDL lengths for all five tree-cut models that can be produced for the thesaurus tree in Figure 1. The best model is the one with the tree-cut [AIRCRAFT, ball, kite, puzzle] indicated by a thick curve in the figure.

3 Clustering Systematic Polysemy

3.1 Generalization Technique

Using the generalization technique in (Li and Abe, 1998) described in the previous section, we wish to extract systematic polysemy automatically from WordNet. Our assumption is that, if a semantic concept is systematically related to another concept, words that have one sense under one concept (sub)tree are likely to have another sense under the other concept (sub)tree. To give an example, Figure 3 shows parts of WordNet noun trees for `ARTIFACT` and `MEASURE`, where subtrees under `CONTAINER` and `CONTAINERFUL` respectively contain "bottle", "bucket" and "spoon". Note a dashed line in the figure indicates an indirect link for more than one level.

Based on this assumption, it seems systematic polysemy in the two trees can be extracted straight-forwardly by clustering each tree according to polysemy as a feature, and by matching of clusters taken from each tree.[6] To this end, the notion of tree-cut and the MDL principle seem to comprise an excellent tool.

However, we must be careful in adopting Li and Abe's technique directly: since the problem which their technique was applied to is fundamentally different from ours, some procedures used in their problem may not have any interpretation in our problem. Although both problems are essentially a tree generalization problem, their problem estimates the true probability distribution from a random sample of examples (a corpus), whereas our problem does not have any additional data to estimate, since all data (a lexicon) is already known. This difference raises the following issue. In the calculation of the data description length in equation (6), each word in a cluster, observed or unobserved, is assigned an estimated probability, which is a uniform fraction of the probability of the cluster. This procedure does not have interpretation if it is applied to our problem. Instead, we use the distribution of feature frequency proportion of the clusters, and calculate the data description length by the following formula:

$$L(S|\Gamma,\Theta) = -\sum_{i=1}^{k} f(C_i) \times log_2 P(C_i) \quad (9)$$

where $\Gamma = [C_1, .., C_k], \Theta = [P(C_1), .., P(C_k)]$. This corresponds to the length required to encode all words in a cluster, for all clusters in a tree-cut, assuming Huffman's algorithm (Huffman, 1952) assigned a codeword of length $-log_2 P(C_i)$ to each cluster C_i (whose propor-

[6] We could also combine two (or possibly more) trees into one tree and apply clustering over that tree once. In this paper, we describe clustering of two trees for example purpose.

Γ	L(Θ|Γ)	L(S|Γ,Θ)	L(M,S)
[A]	1.66	11.60	13.26
[AC,TOY]	3.32	14.34	17.66
[ap,heli,TOY]	4.98	14.44	19.42
[AC,ball,kite,puz]	6.64	4.96	11.60
[ap,hel,ball,kite,puz]	8.31	5.06	13.37

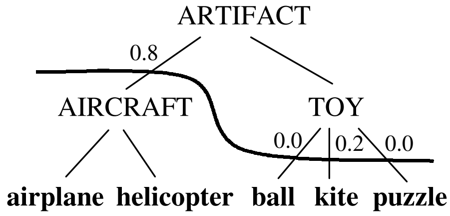

Figure 2: The MDL lengths and the final tree-cut

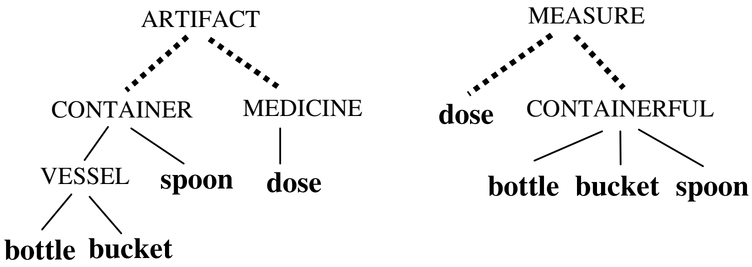

Figure 3: Parts of WordNet trees `ARTIFACT` and `MEASURE`

tion is $P(C_i) = \frac{f(C_i)}{|S|}$).

All other notions and formulas are applicable to our problem without modification.

3.2 Clustering Method

Our clustering method uses the the modified generalization technique described in the last section to generate tree-cuts. But before we apply the method, we must transform the data in Wordnet. This is because WordNet differs from a theaurus tree in two ways: it is a graph rather than a tree, and internal nodes as well as leaf nodes carry data. First, we eliminate multiple inheritance by separating shared subtrees. Second, we bring down every internal node to a leaf level by creating a new duplicate node and adding it as a child of the old node (thus making the old node an internal node).

After trees are transformed, our method extracts systematic polysemy by the following three steps. In the first step, all leaf nodes of the two trees are marked with either 1 or 0 (1 if a node/word appears in both trees, or 0 otherwise).

In the second step, the generalization technique is applied to each tree, and two tree-cuts are obtained. To search for the best tree-cut, instead of computing the description length for all possible tree-cuts in a tree, a greedy dynamic programming algorithm is used. This algorithm, called *Find-MDL* in (Li and Abe, 1998), finds the best tree-cut for a tree by recursively finding the best tree-cuts for all of its subtrees and merging them from bottom up. This algorithm is quite efficient, since it is basically a depth-first search with minor overhead for computing the description length.

Finally in the third step, clusters from the two tree-cuts are matched up, and the pairs which have substantial overlap are selected as systematic polysemy.

Figure 4 shows parts of the final tree-cuts for `ARTIFACT` and `MEASURE` obtained by our method.[7] In both trees, most of the clusters in the tree-cuts are from nodes at depth 1 (counting the root as depth 0). That is because the tree-cut technique used in our method is sensitive to the structure of the tree. More specifically, the MDL principle inherently penalizes a complex tree-cut by assigning a long parameter length. Therefore, unless the entropy of the feature distribution is large enough to make the data length overshadow the parameter length, simpler tree-cuts partitioned at abstract levels are preferred. This situation tends to happen often when the tree is bushy and the total feature frequency is low. This was precisely the case with `ARTIFACT` and `MEASURE`, where both

[7] In the figure, bold letters indicate words which are polysemous in the two tree.

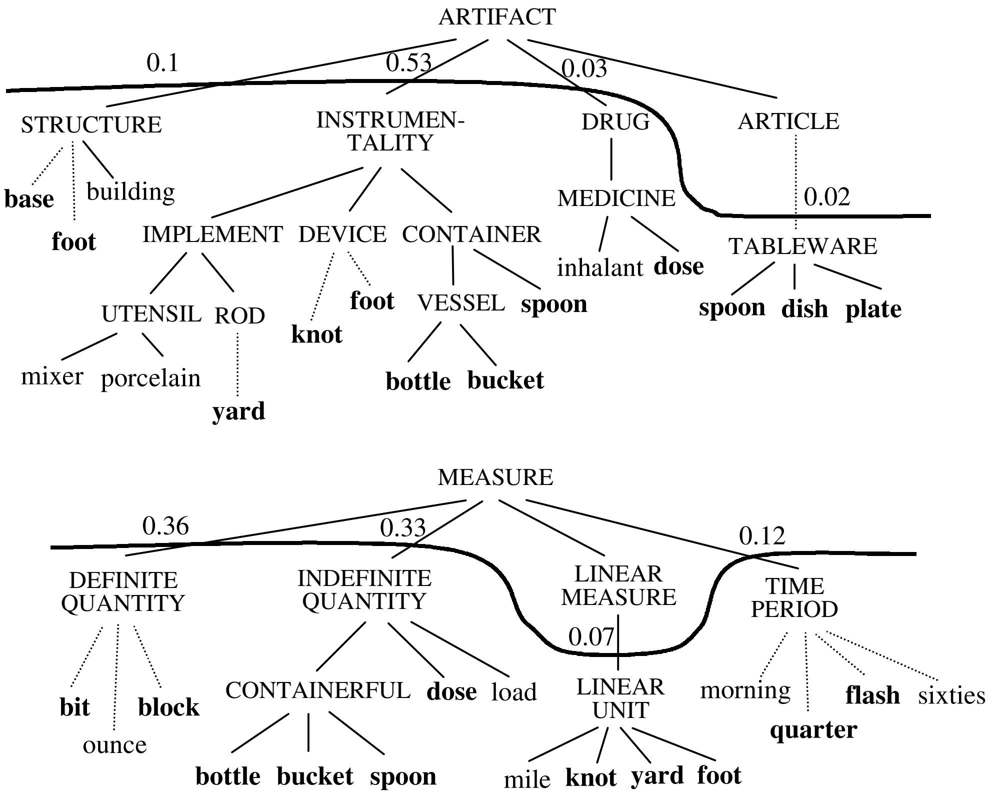

Figure 4: Parts of the final tree-cuts for ARTIFACT and MEASURE

trees were quite bushy, and only 4% and 14% of the words were polysemous in the two categories respectively.

4 Evaluation

To test our method, we chose 5 combinations from WordNet noun Top categories (which we call *top relation classes*), and extracted cluster pairs which have more than 3 overlapping words. Then we evaluated those pairs in two aspects: related vs. unrelated relations, and automatic vs. manual clusters.

4.1 Related vs. Unrelated Clusters

Of the cluster pairs we extracted automatically, not all are systematically related; some are unrelated, homonymous relations. They are essentially false positives for our purposes. Table 1 shows the number of related and unrelated relations in the extracted cluster pairs.

Although the results vary among category combinations, the ratio of the related pairs is rather low: less than 60% on average. There are several reasons for this. First, there are some pairs whose relations are spurious. For example, in ARTIFACT-GROUP class, a pair [LUMBER, SOCIAL_GROUP] was extracted. Words which are common in the two clusters are "picket", "board" and "stock". This relation is obviously homonymous.

Second, some clusters obtained by tree-cut are rather abstract, so that pairing two abstract clusters results in an unrelated pair. For example, in ARTIFACT-MEASURE class, a pair [INSTRUMENTALITY, LINEAR_UNIT] was selected. Words which are common in the two clusters include "yard", "foot" and "knot" (see the previous Figure 4). Here, the concept INSTRUMENTALITY is very general (at depth 1), and it also contains many (polysemous) words. So, matching this cluster with another abstract cluster is likely to yield a pair which has just enough overlapping words but whose relation is not systematic. In the case of [INSTRUMENTALITY, LINEAR_UNIT], the situation is even worse, because the concept of LINEAR_UNIT in MEASURE represents a collection of terms that were chosen arbitrarily in the his-

Table 1: Related vs. Unrelated Relations

Top relation class	Related	Unrelated	Total	% of related
ACTION-LOCATION	10	1	11	90.9
ARTIFACT-GROUP	18	9	27	66.7
ARTIFACT-MEASURE	7	19	26	26.9
ARTIFACT-SUBSTANCE	19	12	31	61.3
COMMUNICATION-PERSON	12	11	23	52.2
Total	66	52	118	55.9

tory of the English language.

4.2 Automatic vs. Manual Clusters

To compare the cluster pairs our method extracted automatically to manually extracted clusters, we use WordNet *cousins*. A cousin relation is relatively new in WordNet, and the coverage is still incomplete. However, it gives us a good measure to see whether our automatic method discovered systematic relations that correspond to human intuitions.

A cousin relation in WordNet is defined between two synsets (currently in the noun trees only), and it indicates that senses of a word that appear in both of the (sub)trees rooted by those synsets are related.[8] The cousins were manually extracted by the WordNet lexicographers. Table 2 shows the number of cousins listed for each top relation class and the number of cousins our automatic method recovered (in the 'Auto' column). As you see, the total recall ratio is over 80% (27/33≈ .82).

In the right three columns of Table 2, we also show the breakdown of the recovered cousins, whether each recovered one was an exact match, or it was more general or specific than the corresponding WordNet cousin. From this, we can see that more than half of the recovered cousins were more general than the WordNet cousins. That is partly because some WordNet cousins have only one or two common words. For example, a WordNet cousin [PAINTING, COLORING_MATERIAL] in ARTIFACT-SUBSTANCE has only one common word "watercolor". Such a minor relation tends to be lost in our tree generalization procedure. However, the main reason is the difficulty mentioned earlier in the paper: the problem of applying the tree-cut technique to a bushy tree when the data is sparse.

In addition to the WordNet cousins, our automatic extraction method discovered several interesting relations. Table 3 shows some examples.

5 Conclusions and Future Work

In this paper, we proposed an automatic method for extracting systematic polysemy from WordNet. As we reported, preliminary results show that our method identified almost all WordNet cousins as well as some new ones. One difficulty is that applying the generalization technique using the MDL principle to the bushy WordNet trees seems to yield a tree-cut at rather abstract level.

For future work, we plan to compare the systematic relations extracted by our automatic method to corpus data. In particular, we like to test whether our method extracts the same groups of senses which human annotators disagreed (Ng *et al.*, 1999). We also like to test whether our method agrees with the finding that multiple senses which occur in a discourse are often systematically polysemous (Krovetz, 1998).

[8]Actually, cousin is one of the three relations which indicate the grouping of related senses of a word. Others are *sister* and *twin*. In this paper, we use cousin to refer to all relations listed in "cousin.tps" file (available in a WordNet distribution).

References

Apresjan, J. (1973). Regular Polysemy. *Linguistics*, (142).

Buitelaar, P. (1997). A Lexicon for Underspecified Semantic Tagging. In *Proceedings of the ACL SIGLEX Workshop on Tagging Text*

Table 2: Automatic Clusters vs. WordNet Cousins

Top relation class	WN cousin	Auto	Exact	Gen	Spec
ACTION-LOCATION	2	1	0	1	0
ARTIFACT-GROUP	6	6	1	5	0
ARTIFACT-MEASURE	1	1	0	1	0
ARTIFACT-SUBSTANCE	15	13	3	9	1
COMMUNICATION-PERSON	9	6	5	1	0
Total	33	27	9	17	1

Table 3: Examples of Automatically Extracted Systematic Polysemy

Top relation class	Relation	Common Words
ACTION-LOCATION	[ACTION, POINT]	"drop", "circle", "intersection", "dig", "crossing", "bull's eye"
ARTIFACT-GROUP	[STRUCTURE, PEOPLE]	"house", "convent", "market", "center"
ARTIFACT-SUBSTANCE	[FABRIC, CHEMICAL_COMPOUND]	"acetate", "nylon", "acrylic", "polyester"
COMMUNICATION-PERSON	[VOICE, SINGER]	"soprano", "alto", "tenor", "baritone"
	[WRITING, RELIGIOUS_PERSON]	"John", "Matthew", "Jonah", "Joshua", "Jeremiah"

with Lexical Semantics, Washington, D.C., pp. 25-33.

Buitelaar, P. (1998). CORELEX: Systematic Polysemy and Underspecification. Ph.D. dissertation, Department of Computer Science, Brandeis University.

Copestake, A. and Briscoe, T. (1995). Semi-productive Polysemy and Sense Extension. *Journal of Semantics*, **12**.

Cruse, D. (1986). *Lexical Semantics*, Cambridge University Press.

Huffman, D. A. (1952). A Model for the Construction of Minimum Redundancy Codes. In *Proceedings of the IRE*, **40**.

Kilgarriff, A. (1998a). SENSEVAL: An Exercise in Evaluating Word Sense Disambiguation Programs. In *Proceedings of the LREC*

Kilgarriff, A. (1998b). Inter-tagger Agreement. In *Advanced Papers of the SENSEVAL Workshop*, Sussex, UK.

Krovetz, R. (1998). More than One Sense Per Discourse. In *Advanced Papers of the SENSEVAL Workshop*, Sussex, UK.

Li, H. and Abe, N. (1998). Generalizing Case Frames Using a Thesaurus and the MDL Principle, *Computational Linguistics*, 24(2), pp. 217-244

Miller, G. (eds.) (1990). WORDNET: An Online Lexical Database. *International Journal of Lexicography*, **3** (4).

Ng, H.T., Lim, C. and Foo, S. (1999). A Case Study on Inter-Annotator Agreement for Word Sense Disambiguation. In *Proceedings of the ACL SIGLEX Workshop on Standardizing Lexical Resources*, College Park, MD.

Nunberg, G. (1995). Transfers of Meaning. *Journal of Semantics*, **12**.

Procter, P. (1978). *Longman dictionary of Contemporary English*, Longman Group.

Pustejovsky, J. (1995). *The Generative Lexicon*, The MIT Press.

Rissanen, J. (1978). Modeling by Shortest Data Description. *Automatic*, **14**.

Veronis, J. (1998). A Study of Polysemy Judgements and Inter-annotator Agreement. In *Advanced Papers of the SENSEVAL Workshop*, Sussex, UK.

Vossen, P., Peters, W. and Gonzalo, J. (1999). Towards a Universal Index of Meaning. In *Proceedings of the ACL SIGLEX Workshop on Standardizing Lexical Resources*, College Park, MD.