Syntactic Features and Word Similarity for Supervised Metonymy Resolution

Malvina Nissim
ICCS, School of Informatics
University of Edinburgh
mnissim@inf.ed.ac.uk

Katja Markert
ICCS, School of Informatics
University of Edinburgh and
School of Computing
University of Leeds
markert@inf.ed.ac.uk

Abstract

We present a supervised machine learning algorithm for metonymy resolution, which exploits the similarity between examples of conventional metonymy. We show that syntactic head-modifier relations are a high precision feature for metonymy recognition but suffer from data sparseness. We partially overcome this problem by integrating a thesaurus and introducing simpler grammatical features, thereby preserving precision and increasing recall. Our algorithm generalises over two levels of contextual similarity. Resulting inferences exceed the complexity of inferences undertaken in word sense disambiguation. We also compare automatic and manual methods for syntactic feature extraction.

1 Introduction

Metonymy is a figure of speech, in which one expression is used to refer to the standard referent of a related one (Lakoff and Johnson, 1980). In (1),[1] "seat 19" refers to the person occupying seat 19.

(1) *Ask **seat 19** whether he wants to swap*

The importance of resolving metonymies has been shown for a variety of NLP tasks, e.g., machine translation (Kamei and Wakao, 1992), question answering (Stallard, 1993) and anaphora resolution (Harabagiu, 1998; Markert and Hahn, 2002).

In order to recognise and interpret the metonymy in (1), a large amount of knowledge and contextual inference is necessary (e.g. seats cannot be questioned, people occupy seats, people can be questioned). Metonymic readings are also potentially open-ended (Nunberg, 1978), so that developing a machine learning algorithm based on previous examples does not seem feasible.

However, it has long been recognised that many metonymic readings are actually quite regular (Lakoff and Johnson, 1980; Nunberg, 1995).[2] In (2), "Pakistan", the name of a location, refers to one of its national sports teams.[3]

(2) ***Pakistan** had won the World Cup*

Similar examples can be regularly found for many other location names (see (3) and (4)).

(3) ***England** won the World Cup*

(4) ***Scotland** lost in the semi-final*

In contrast to (1), the regularity of these examples can be exploited by a supervised machine learning algorithm, although this method is not pursued in standard approaches to regular polysemy and metonymy (with the exception of our own previous work in (Markert and Nissim, 2002a)). Such an algorithm needs to infer from examples like (2) (when labelled as a metonymy) that "England" and "Scotland" in (3) and (4) are also metonymic. In order to

[1] (1) was actually uttered by a flight attendant on a plane.

[2] Due to its regularity, conventional metonymy is also known as *regular polysemy* (Copestake and Briscoe, 1995). We use the term "metonymy" to encompass both conventional and unconventional readings.

[3] All following examples are from the British National Corpus (BNC, http://info.ox.ac.uk/bnc).

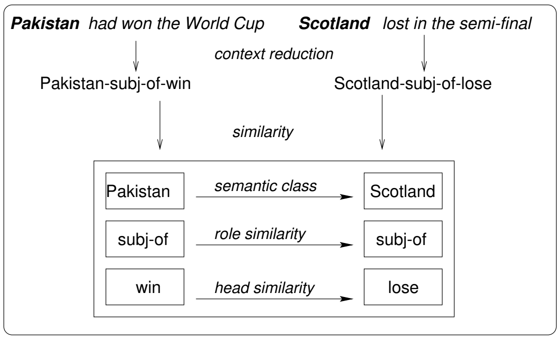

Figure 1: Context reduction and similarity levels

draw this inference, two levels of similarity need to be taken into account. One concerns the similarity of the words to be recognised as metonymic or literal (*Possibly Metonymic Words*, PMWs). In the above examples, the PMWs are "Pakistan", "England" and "Scotland". The other level pertains to the similarity between the PMW's contexts (*"<subject> (had) won the World Cup"* and *"<subject> lost in the semi-final"*). In this paper, we show how a machine learning algorithm can exploit both similarities.

Our corpus study on the *semantic class* of locations confirms that regular *metonymic patterns*, e.g., using a place name for any of its sports teams, cover most metonymies, whereas unconventional metonymies like (1) are very rare (Section 2). Thus, we can recast metonymy resolution as a classification task operating on semantic classes (Section 3).

In Section 4, we restrict the classifier's features to head-modifier relations involving the PMW. In both (2) and (3), the context is reduced to *subj-of-win*. This allows the inference from (2) to (3), as they have the *same* feature value. Although the remaining context is discarded, this feature achieves high precision. In Section 5, we generalize context similarity to draw inferences from (2) or (3) to (4). We exploit both the similarity of the heads in the grammatical relation (e.g., "win" and "lose") and that of the grammatical role (e.g. subject). Figure 1 illustrates context reduction and similarity levels.

We evaluate the impact of automatic extraction of head-modifier relations in Section 6. Finally, we discuss related work and our contributions.

2 Corpus Study

We summarize (Markert and Nissim, 2002b)'s annotation scheme for location names and present an annotated corpus of occurrences of country names.

2.1 Annotation Scheme for Location Names

We identify *literal*, *metonymic*, and *mixed* readings.

The `literal` reading comprises a locative (5) and a political entity interpretation (6).

(5) *coral coast of* ***Papua New Guinea***

(6) ***Britain****'s current account deficit*

We distinguish the following metonymic patterns (see also (Lakoff and Johnson, 1980; Fass, 1997; Stern, 1931)). In a `place-for-people` pattern, a place stands for any persons/organisations associated with it, e.g., for sports teams in (2), (3), and (4), and for the government in (7).[4]

(7) *a cardinal element in* ***Iran****'s strategy when Iranian naval craft [...] bombarded [...]*

In a `place-for-event` pattern, a location name refers to an event that occurred there (e.g., using the word Vietnam for the Vietnam war). In a `place-for-product` pattern a place stands for a product manufactured there (e.g., the word Bordeaux referring to the local wine).

The category `othermet` covers unconventional metonymies, as (1), and is only used if none of the other categories fits (Markert and Nissim, 2002b).

We also found examples where two predicates are involved, each triggering a different reading.

(8) *they arrived in* ***Nigeria****, hitherto a leading critic of the South African regime*

In (8), both a literal (triggered by "arriving in") and a `place-for-people` reading (triggered by "leading critic") are invoked. We introduced the category `mixed` to deal with these cases.

2.2 Annotation Results

Using Gsearch (Corley et al., 2001), we randomly extracted 1000 occurrences of country names from the BNC, allowing any country name and its variants listed in the CIA factbook[5] or WordNet (Fellbaum,

[4] As the explicit referent is often underspecified, we introduce `place-for-people` as a *supertype* category and we evaluate our system on supertype classification in this paper. In the annotation, we further specify the different groups of people referred to, whenever possible (Markert and Nissim, 2002b).

[5] `http://www.cia.gov/cia/publications/factbook/`

1998) to occur. Each country name is surrounded by three sentences of context.

The 1000 examples of our corpus have been independently annotated by two computational linguists, who are the authors of this paper. The annotation can be considered reliable (Krippendorff, 1980) with 95% agreement and a *kappa* (Carletta, 1996) of .88. Our corpus for testing and training the algorithm includes only the examples which both annotators could agree on and which were not marked as *noise* (e.g. homonyms, as *"Professor **Greenland**"*), for a total of 925. Table 1 reports the reading distribution.

Table 1: Distribution of readings in our corpus

reading	freq	%
`literal`	737	79.7
`place-for-people`	161	17.4
`place-for-event`	3	.3
`place-for-product`	0	.0
`mixed`	15	1.6
`othermet`	9	1.0
total non-literal	188	20.3
total	925	100.0

3 Metonymy Resolution as a Classification Task

The corpus distribution confirms that metonymies that do not follow established metonymic patterns (`othermet`) are very rare. This seems to be the case for other kinds of metonymies, too (Verspoor, 1997). We can therefore reformulate metonymy resolution as a *classification task* between the literal reading and a fixed set of metonymic patterns that can be identified in advance for particular semantic classes. This approach makes the task comparable to classic *word sense disambiguation* (WSD), which is also concerned with distinguishing between possible word senses/interpretations.

However, whereas a classic (supervised) WSD algorithm is trained on a set of labelled instances of *one particular word* and assigns word senses to new test instances of the *same word*, (supervised) metonymy recognition can be trained on a set of labelled instances of *different words of one semantic class* and assign literal readings and metonymic patterns to new test instances of *possibly different words of the same semantic class*. This class-based approach enables one to, for example, infer the reading of (3) from that of (2).

We use a decision list (DL) classifier. All features encountered in the training data are ranked in the DL (best evidence first) according to the following log-likelihood ratio (Yarowsky, 1995):

$$Log\left(\frac{Pr(reading_i|feature_k)}{\sum_{j\neq i} Pr(reading_j|feature_k)}\right)$$

We estimated probabilities via maximum likelihood, adopting a simple smoothing method (Martinez and Agirre, 2000): 0.1 is added to both the denominator and numerator.

The target readings to be distinguished are `literal`, `place-for-people`, `place-for-event`, `place-for-product`, `othermet` and `mixed`. All our algorithms are tested on our annotated corpus, employing 10-fold cross-validation. We evaluate accuracy and coverage:

$$Acc = \frac{\text{\# correct decisions made}}{\text{\# decisions made}}$$

$$Cov = \frac{\text{\# decisions made}}{\text{\# test data}}$$

We also use a backing-off strategy to the most frequent reading (`literal`) for the cases where no decision can be made. We report the results as accuracy backoff (Acc_b); coverage backoff is always 1. We are also interested in the algorithm's performance in recognising non-literal readings. Therefore, we compute precision (P), recall (R), and F-measure (F), where A is the number of non-literal readings correctly identified as non-literal (true positives) and B the number of literal readings that are incorrectly identified as non-literal (false positives):

$$P = A/(A+B)$$

$$R = \frac{A}{\text{\#non-literal examples in the test data}}$$

$$F = 2PR/(R+P)$$

The baseline used for comparison is the assignment of the most frequent reading `literal`.

4 Context Reduction

We show that reducing the context to head-modifier relations involving the Possibly Metonymic Word achieves high precision metonymy recognition.[6]

[6]In (Markert and Nissim, 2002a), we also considered local and topical cooccurrences as contextual features. They constantly achieved lower precision than grammatical features.

Table 2: Example feature values for role-of-head

role-of-head (*r-of-h*)	example
subj-of-win	***England*** *won the World Cup* (place-for-people)
subjp-of-govern	***Britain*** *has been governed by ...* (literal)
dobj-of-visit	*the Apostle had visited* ***Spain*** (literal)
gen-of-strategy	*in* ***Iran****'s strategy ...* (place-for-people)
premod-of-veteran	*a* ***Vietnam*** *veteran from Rhode Island* (place-for-event)
ppmod-of-with	*its border with* ***Hungary*** (literal)

Table 3: Role distribution

role	freq	#non-lit
subj	92	65
subjp	6	4
dobj	28	12
gen	93	20
premod	94	13
ppmod	522	57
other	90	17
total	925	188

We represent each example in our corpus by a single feature role-of-head, expressing the grammatical role of the PMW (limited to (active) subject, passive subject, direct object, modifier in a prenominal genitive, other nominal premodifier, dependent in a prepositional phrase) and its lemmatised lexical head within a dependency grammar framework.[7] Table 2 shows example values and Table 3 the role distribution in our corpus.

We trained and tested our algorithm with this feature (hmr).[8] Results for hmr are reported in the first line of Table 5. The reasonably high precision (74.5%) and accuracy (90.2%) indicate that reducing the context to a head-modifier feature does not cause loss of crucial information in most cases. Low recall is mainly due to low coverage (see Problem 2 below). We identified two main problems.

Problem 1. The feature *can* be too simplistic, so that decisions based on the head-modifier relation can assign the wrong reading in the following cases:

- "Bad" heads: Some lexical heads are semantically empty, thus failing to provide strong evidence for any reading and lowering both recall and precision. Bad predictors are the verbs "to have" and "to be" and some prepositions such as "with", which can be used with metonymic (*talk with* ***Hungary***) and literal (*border with* ***Hungary***) readings. This problem is more serious for function than for content word heads: precision on the set of subjects and objects is 81.8%, but only 73.3% on PPs.
- "Bad" relations: The *premod* relation suffers from noun-noun compound ambiguity. ***US*** *operation* can refer to an operation *in* the US (literal) or *by* the US (metonymic).
- Other cases: Very rarely neglecting the remaining context leads to errors, even for "good" lexical heads and relations. Inferring from the metonymy in (4) that "Germany" in *"**Germany** lost a fifth of its territory"* is also metonymic, e.g., is wrong and lowers precision.

However, wrong assignments (based on head-modifier relations) do not constitute a major problem as accuracy is very high (90.2%).

Problem 2. The algorithm is often unable to make any decision that is based on the head-modifier relation. This is by far the more frequent problem, which we adress in the remainder of the paper. The feature role-of-head accounts for the similarity between (2) and (3) only, as classification of a test instance with a particular feature value relies on having seen *exactly the same* feature value in the training data. Therefore, we have not tackled the inference from (2) or (3) to (4). This problem manifests itself in data sparseness and low recall and coverage, as many heads are encountered only once in the corpus. As hmr's coverage is only 63.1%, backoff to a literal reading is required in 36.9% of the cases.

5 Generalising Context Similarity

In order to draw the more complex inference from (2) or (3) to (4) we need to generalise context similarity. We relax the identity constraint of the original algorithm (the *same* role-of-head value of the test instance must be found in the DL), exploiting two similarity levels. Firstly, we allow to draw inferences over similar values of lexical heads (e.g. from *subj-of-win* to *subj-of-lose*), rather than over identical ones only. Secondly, we allow to discard the

[7] We consider only one link per PMW, although cases like (8) would benefit from including all links the PMW participates in.

[8] The feature values were manually annotated for the following experiments, adapting the guidelines in (Poesio, 2000). The effect of automatic feature extraction is described in Section 6.

Table 4: Example thesaurus entries

lose[V]:	**win**$_1$ **0.216**, gain$_2$ 0.209, have$_3$ 0.207, ...
attitude[N]:	stance$_1$ 0.181, behavior$_2$ 0.18, ..., **strategy**$_{17}$ **0.128**

lexical head and generalise over the PMW's grammatical role (e.g. subject). These generalisations allow us to double recall without sacrificing precision or increasing the size of the training set.

5.1 Relaxing Lexical Heads

We regard two feature values *r-of-h* and *r-of-h*$'$ as similar if h and h' are similar. In order to capture the similarity between h and h' we integrate a thesaurus (Lin, 1998) in our algorithm's testing phase. In Lin's thesaurus, similarity between words is determined by their distribution in dependency relations in a newswire corpus. For a content word h (e.g., "lose") of a specific part-of-speech a set of similar words Σ_h of the same part-of-speech is given. The set members are ranked in decreasing order by a similarity score. Table 4 reports example entries.[9]

Our modified algorithm (relax I) is as follows:

1. train DL with role-of-head as in hmr; for each test instance observe the following procedure (*r-of-h* indicates the feature value of the test instance);
2. **if** *r-of-h* is found in the DL, apply the corresponding rule and stop;

2$'$ **otherwise** choose a number $n \geq 1$ and set $i = 1$;

   (a) extract the i^{th} most similar word h_i to h from the thesaurus;

   (b) **if** $i > n$ or the similarity score of $h_i < 0.10$, assign no reading and stop;

   (b') **otherwise**: if *r-of-*h_i is found in the DL, apply corresponding rule and stop; if *r-of-*h_i is not found in the DL, increase i by 1 and go to (a);

The examples already covered by hmr are classified in exactly the same way by relax I (see Step 2). Let us therefore assume we encounter the test instance (4), its feature value *subj-of-lose* has not been seen in the training data (so that Step 2 fails and Step 2$'$ has to be applied) and *subj-of-win* is in the DL. For all $n \geq 1$, relax I will use the rule for *subj-of-win* to assign a reading to "Scotland" in (4) as "win" is the most similar word to "lose" in the thesaurus (see Table 4). In this case (2b') is only applied once as already the first iteration over the thesaurus finds a word h_1 with *r-of-*h_1 in the DL.

[9]In the original thesaurus, each Σ_h is subdivided into clusters. We do not take these divisions into account.

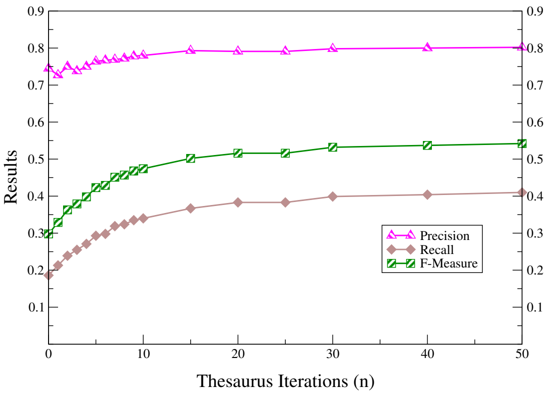

Figure 2: Results for relax I

The classification of "Turkey" with feature value *gen-of-attitude* in (9) required 17 iterations to find a word h_{17} ("strategy"; see Example (7)) similar to "attitude", with *r-of-*h_{17} (*gen-of-strategy*) in the DL.

(9) *To say that this sums up* ***Turkey****'s attitude as a whole would nevertheless be untrue*

Precision, recall and F-measure for $n \in \{1, ..., 10, 15, 20, 25, 30, 40, 50\}$ are visualised in Figure 2. Both precision and recall increase with n. Recall more than doubles from 18.6% in hmr to 41% and precision increases from 74.5% in hmr to 80.2%, yielding an increase in F-measure from 29.8% to 54.2% ($n = 50$). Coverage rises to 78.9% and accuracy backoff to 85.1% (Table 5).

Whereas the increase in coverage and recall is quite intuitive, the high precision achieved by relax I requires further explanation. Let *S* be the set of examples that relax I covers. It consists of two subsets: *S1* is the subset already covered by hmr and its treatment does not change in relax I, yielding the same precision. *S2* is the set of examples that relax I covers *in addition to* hmr. The examples in *S2* consist of cases with highly predictive content word heads as (a) function words are not included in the thesaurus and (b) unpredictive content word heads like "have" or "be" are very frequent and normally already covered by hmr (they are therefore members of *S1*). Precision on *S2* is very high (84%) and raises the overall precision on the set *S*.

Cases that relax I does not cover are mainly due to (a) missing thesaurus entries (e.g., many proper

Table 5: Results summary for manual annotation. For relax I and combination we report best results (50 thesaurus iterations).

algorithm	Acc	Cov	Acc_b	P	R	F
hmr	**.902**	.631	.817	**.745**	.186	.298
relax I	.877	.789	**.851**	**.802**	.410	**.542**
relax II	.865	**.903**	**.859**	**.813**	.441	**.572**
combination	.894	**.797**	**.870**	**.814**	**.510**	**.627**
baseline	.797	1.00	.797	n/a	.000	n/a

names or alternative spelling), (b) the small number of training instances for some grammatical roles (e.g. *dobj*), so that even after 50 thesaurus iterations no similar role-of-head value could be found that is covered in the DL, or (c) grammatical roles that are not covered (*other* in Table 3).

5.2 Discarding Lexical Heads

Another way of capturing the similarity between (3) and (4), or (7) and (9) is to ignore lexical heads and generalise over the grammatical role (role) of the PMW (with the feature values as in Table 3: *subj*, *subjp*, *dobj*, *gen*, *premod*, *ppmod*). We therefore developed the algorithm relax II.

1. train decision lists:
   (a) DL1 with role-of-head as in hmr
   (b) DL2 with role;

   for each test instance observe the following procedure (*r-of-h* and *r* are the feature values of the test instance);
2. **if** *r-of-h* is found in the DL1, apply the corresponding rule and stop;

2' **otherwise**, if *r* is found in DL2, apply the corresponding rule.

Let us assume we encounter the test instance (4), *subj-of-lose* is not in DL1 (so that Step 2 fails and Step $2'$ has to be applied) and *subj* is in DL2. The algorithm relax II will assign a `place-for-people` reading to "Scotland", as most subjects in our corpus are metonymic (see Table 3).

Generalising over the grammatical role outperforms hmr, achieving 81.3% precision, 44.1% recall, and 57.2% F-measure (see Table 5). The algorithm relax II also yields fewer false negatives than relax I (and therefore higher recall) since all subjects not covered in DL1 are assigned a metonymic reading, which is not true for relax I.

5.3 Combining Generalisations

There are several ways of combining the algorithms we introduced. In our experiments, the most successful one exploits the facts that relax II performs better than relax I on subjects and that relax I performs better on the other roles. Therefore the algorithm combination uses relax II if the test instance is a subject, and relax I otherwise. This yields the best results so far, with 87% accuracy backoff and 62.7% F-measure (Table 5).

6 Influence of Parsing

The results obtained by training and testing our classifier with manually annotated grammatical relations are the upper bound of what can be achieved by using these features. To evaluate the influence parsing has on the results, we used the RASP toolkit (Briscoe and Carroll, 2002) that includes a pipeline of tokenisation, tagging and state-of-the-art statistical parsing, allowing multiple word tags. The toolkit also maps parse trees to representations of grammatical relations, which we in turn could map in a straightforward way to our role categories.

RASP produces at least partial parses for 96% of our examples. However, some of these parses do not assign any role of our roleset to the PMW — only 76.9% of the PMWs are assigned such a role by RASP (in contrast to 90.2% in the manual annotation; see Table 3). RASP recognises PMW subjects with 79% precision and 81% recall. For PMW direct objects, precision is 60% and recall 86%.[10]

We reproduced all experiments using the automatically extracted relations. Although the relative performance of the algorithms remains mostly unchanged, most of the resulting F-measures are more than 10% lower than for hand annotated roles (Table 6). This is in line with results in (Gildea and Palmer, 2002), who compare the effect of manual and automatic parsing on semantic predicate-argument recognition.

7 Related Work

Previous Approaches to Metonymy Recognition. Our approach is the first machine learning algorithm to metonymy recognition, building on our previous

[10]We did not evaluate RASP's performance on relations that do not involve the PMW.

Table 6: Results summary for the different algorithms using RASP. For relax I and combination we report best results (50 thesaurus iterations).

algorithm	Acc	Cov	Acc_b	P	R	F
hmr	.884	.514	.812	.674	.154	.251
relax I	.841	.666	.821	.619	.319	.421
relax II	.820	.769	.823	.621	.340	.439
combination	.850	.672	.830	.640	.388	.483
baseline	.797	1.00	.797	n/a	.000	n/a

work (Markert and Nissim, 2002a). The current approach expands on it by including a larger number of grammatical relations, thesaurus integration, and an assessment of the influence of parsing. Best F-measure for manual annotated roles increased from 46.7% to 62.7% on the same dataset.

Most other traditional approaches rely on hand-crafted knowledge bases or lexica and use *violations* of hand-modelled selectional restrictions (plus sometimes syntactic violations) for metonymy recognition (Pustejovsky, 1995; Hobbs et al., 1993; Fass, 1997; Copestake and Briscoe, 1995; Stallard, 1993).[11] In these approaches, selectional restrictions (SRs) are not seen as preferences but as absolute constraints. If and only if such an absolute constraint is violated, a non-literal reading is proposed. Our system, instead, does not have *any* a priori knowledge of semantic predicate-argument restrictions. Rather, it refers to previously seen training examples in head-modifier relations and their labelled senses and computes the likelihood of each sense using this distribution. This is an advantage as our algorithm also resolved metonymies *without* SR violations in our experiments. An empirical comparison between our approach in (Markert and Nissim, 2002a)[12] and an SRs violation approach showed that our approach performed better.

In contrast to previous approaches (Fass, 1997; Hobbs et al., 1993; Copestake and Briscoe, 1995; Pustejovsky, 1995; Verspoor, 1996; Markert and Hahn, 2002; Harabagiu, 1998; Stallard, 1993), we use a corpus reliably annotated for metonymy for evaluation, moving the field towards more objective evaluation procedures.

Word Sense Disambiguation. We compared our approach to supervised WSD in Section 3, stressing word-to-word vs. class-to-class inference. This allows for a level of abstraction not present in standard supervised WSD. We can infer readings for words that have not been seen in the training data before, allow an easy treatment of rare words that undergo regular sense alternations and do not have to annotate and train separately for every individual word to treat regular sense distinctions.[13]

By exploiting additional similarity levels and integrating a thesaurus we further generalise the kind of inferences we can make and limit the size of annotated training data: as our sampling frame contains 553 different names, an annotated data set of 925 samples is quite small. These generalisations over context and collocates are also applicable to standard WSD and can supplement those achieved e.g., by subcategorisation frames (Martinez et al., 2002). Our approach to word similarity to overcome data sparseness is perhaps most similar to (Karov and Edelman, 1998). However, they mainly focus on the computation of similarity measures from the training data. We instead use an off-the-shelf resource without adding much computational complexity and achieve a considerable improvement in our results.

8 Conclusions

We presented a supervised classification algorithm for metonymy recognition, which exploits the similarity between examples of conventional metonymy, operates on semantic classes and thereby enables complex inferences from training to test examples. We showed that syntactic head-modifier relations are a high precision feature for metonymy recognition. However, basing inferences only on the lexical heads seen in the training data leads to data sparseness due to the large number of different lexical heads encountered in natural language texts. In order to overcome this problem we have integrated a thesaurus that allows us to draw inferences be-

[11](Markert and Hahn, 2002) and (Harabagiu, 1998) enhance this with anaphoric information. (Briscoe and Copestake, 1999) propose using frequency information besides syntactic/semantic restrictions, but use only a priori sense frequencies without contextual features.

[12]Note that our current approach even outperforms (Markert and Nissim, 2002a).

[13]Incorporating knowledge about particular PMWs (e.g., as a prior) will probably improve performance, as word idiosyncracies — which can still exist even when treating regular sense distinctions — could be accounted for. In addition, knowledge about the individual word is necessary to assign its original semantic class.

tween examples with similar but not identical lexical heads. We also explored the use of simpler grammatical role features that allow further generalisations. The results show a substantial increase in precision, recall and F-measure. In the future, we will experiment with combining grammatical features and local/topical cooccurrences. The use of semantic classes and lexical head similarity generalises over two levels of contextual similarity, which exceeds the complexity of inferences undertaken in standard supervised word sense disambiguation.

Acknowledgements. The research reported in this paper was supported by ESRC Grant R000239444. Katja Markert is funded by an Emmy Noether Fellowship of the Deutsche Forschungsgemeinschaft (DFG). We thank three anonymous reviewers for their comments and suggestions.

References

E. Briscoe and J. Carroll. 2002. Robust accurate statistical annotation of general text. In *Proc. of LREC, 2002*, pages 1499–1504.

T. Briscoe and A. Copestake. 1999. Lexical rules in constraint-based grammar. *Computational Linguistics*, 25(4):487–526.

J. Carletta. 1996. Assessing agreement on classification tasks: The kappa statistic. *Computational Linguistics*, 22(2):249–254.

A. Copestake and T. Briscoe. 1995. Semi-productive polysemy and sense extension. *Journal of Semantics*, 12:15–67.

S. Corley, M. Corley, F. Keller, M. Crocker, and S. Trewin. 2001. Finding syntactic structure in unparsed corpora: The Gsearch corpus query system. *Computers and the Humanities*, 35(2):81–94.

D. Fass. 1997. *Processing Metaphor and Metonymy*. Ablex, Stanford, CA.

C. Fellbaum, ed. 1998. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, Mass.

D. Gildea and M. Palmer. 2002. The necessity of parsing for predicate argument recognition. In *Proc. of ACL, 2002*, pages 239–246.

S. Harabagiu. 1998. Deriving metonymic coercions from WordNet. In *Workshop on the Usage of WordNet in Natural Language Processing Systems, COLING-ACL, 1998*, pages 142–148.

J. R. Hobbs, M. E. Stickel, D. E. Appelt, and P. Martin. 1993. Interpretation as abduction. *Artificial Intelligence*, 63:69–142.

S. Kamei and T. Wakao. 1992. Metonymy: Reassessment, survey of acceptability and its treatment in machine translation systems. In *Proc. of ACL, 1992*, pages 309–311.

Y. Karov and S. Edelman. 1998. Similarity-based word sense disambiguation. *Computational Linguistics*, 24(1):41-59.

K. Krippendorff. 1980. *Content Analysis: An Introduction to Its Methodology*. Sage Publications.

G. Lakoff and M. Johnson. 1980. *Metaphors We Live By*. Chicago University Press, Chicago, Ill.

D. Lin. 1998. An information-theoretic definition of similarity. In *Proc. of International Conference on Machine Learning, Madison, Wisconsin*.

K. Markert and U. Hahn. 2002. Understanding metonymies in discourse. *Artificial Intelligence*, 135(1/2):145–198.

K. Markert and M. Nissim. 2002a. Metonymy resolution as a classification task. In *Proc. of EMNLP, 2002*, pages 204–213.

Katja Markert and Malvina Nissim. 2002b. Towards a corpus annotated for metonymies: the case of location names. In *Proc. of LREC, 2002*, pages 1385–1392.

D. Martinez and E. Agirre. 2000. One sense per collocation and genre/topic variations. In *Proc. of EMNLP, 2000*.

D. Martinez, E. Agirre, and L. Marquez. 2002. Syntactic features for high precision word sense disambiguation. In *Proc. of COLING, 2002*.

G. Nunberg. 1978. *The Pragmatics of Reference*. Ph.D. thesis, City University of New York, New York.

G. Nunberg. 1995. Transfers of meaning. *Journal of Semantics*, 12:109–132.

M. Poesio, 2000. *The GNOME Annotation Scheme Manual*. University of Edinburgh, 4^{th} version. Available from `http://www.hcrc.ed.ac.uk/~gnome`.

J. Pustejovsky. 1995. *The Generative Lexicon*. MIT Press, Cambridge, Mass.

D. Stallard. 1993. Two kinds of metonymy. In *Proc. of ACL, 1993*, pages 87–94.

G. Stern. 1931. *Meaning and Change of Meaning*. Göteborg: Wettergren & Kerbers Förlag.

C. Verspoor. 1996. Lexical limits on the influence of context. In *Proc. of CogSci, 1996*, pages 116–120.

C. Verspoor. 1997. Conventionality-governed logical metonymy. In H. Bunt et al., editors, *Proc. of IWCS-2, 1997*, pages 300–312.

D. Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proc. of ACL, 1995*, pages 189–196.