Mapping WordNet synsets to Wikipedia articles

Samuel Fernando, Mark Stevenson

Department of Computer Science, University of Sheffield
Regent Court, 211 Portobello, Sheffield, S1 4DP
s.fernando@shef.ac.uk, m.stevenson@dcs.shef.ac.uk

Abstract

Lexical knowledge bases (LKBs), such as WordNet, have been shown to be useful for a range of language processing tasks. Extending these resources is an expensive and time-consuming process. This paper describes an approach to address this problem by automatically generating a mapping from WordNet synsets to Wikipedia articles. A sample of synsets has been manually annotated with article matches for evaluation purposes. The automatic methods are shown to create mappings with precision of 87.8% and recall of 46.9%. These mappings can then be used as a basis for enriching WordNet with new relations based on Wikipedia links. The manual and automatically created data is available online.

Keywords: Aligning lexical resources, WordNet, Wikipedia

1. Introduction

LKBs have been shown to be useful for a wide range of language processing applications. WordNet in particular is the most widely used LKB in current research because of its free availability and wide coverage. WordNet has been used for a wide range of language processing applications including semantic search (Benassi et al., 2004), text summarisation (Carenini et al., 2008) and word sense disambiguation (Agirre and Soroa, 2009).

Despite its popularity WordNet lacks some information that is useful for language processing. For example, it does not connected words which are linked by topic. In WordNet "tennis player" is not related to "racquet", despite the connection between them. However this information would potentially be useful for several applications including Information Retrieval and Word Sense Disambiguation.

The online encyclopedia Wikipedia contains massive amounts of information which may address this problem. For example, the Wikipedia article on the topic "tennis" mentions both "tennis player" and "racquet". However one significant problem is that WordNet and Wikipedia both contain ambiguity. For example, in WordNet "racket" can mean 'loud and disturbing noise', 'illegal enterprise', 'auditory experience that lacks musical quality' or 'sports implement'. In addition, Wikipedia contains several articles with the title 'racket' including ones that refer to a film, a programming language and sports implement. Identifying the Wikipedia article that is associated with each WordNet synset (if there is one) is a key step in making use of the information it contains to enrich WordNet. This problem is addressed in this paper by using mappings methods to match synsets to articles. The resulting mapping is then evaluated against manual annotations.

2. Related work

There have been previous attempts to connect WordNet and Wikipedia. Ruiz-Casado et al. (2005) use text similarity to link articles to synsets. This was done using the Simple English Wikipedia, a much smaller resource than the full Wikipedia used in this paper. Suchanek et al. (2008) uses heuristic methods to link Wikipedia categories to synsets in the WordNet hierarchy. Recent work by Ponzetto and Navigli (2010) maps Wikipedia articles to WordNet synsets, using text overlap methods to find the best match. However only article titles are used to find possible matches, not the article content.

Unlike previous approaches to this problem the method described here attempts to find the best matching article in Wikipedia for each noun synset in WordNet. Mapping in this direction is a significantly different problem since Wikipedia is much larger than WordNet. In addition, unlike the approach described by Ponzetto and Navigli (2010) the methods used here can find matching articles even where the title of the article does not match any of the synset words.

3. Mapping method

The process of mapping WordNet synsets to Wikipedia articles is divided into a 3 stage approach. The first stage (**Generation of Candidate Articles**) aims to reduces the search space by identifying a small (but high recall) set of candidate articles for each noun synset. Two approaches are used, matching words in WordNet synsets against Wikipedia article titles and using an Information Retrieval system to search the full article text. The second stage (**Selecting the Best Mappings**) uses this candidate article set to select the best matching article for each synset (or decide that none of the candidate article represents a good match). Text similarity metrics are used to find the best match. The third stage (**Refining the Mappings**) uses a global scoring approach and Wikipedia links to select a more precise set of mappings. Two types of methods are used. The first eliminates all many-to-1 matches, leaving a 1-to-1 mapping. The second uses Wikipedia links to confirm good matches: matches are only kept where the article links to another mapped article. Additionally a further approach requires that a reciprocal link exists (giving a bi-directional link).

3.1. Generation of Candidate Articles

Two methods were used to find candidate articles: title matching (Section 3.1.1.) and Information Retrieval (Section 3.1.2.).

3.1.1. Title Matching

The title matching approach examines the titles of Wikipedia articles to identify WordNet synsets that could map onto them. Each noun synset S contains several synonymous words, w_1, w_2, ... w_n. For each word w_i in S, a search is carried out in Wikipedia and all articles returned by the search added to the set of candidate articles, C.

1. **Articles** Articles whose title matches w_i are added to C. For example, if w_i is "automobile" the article 'Automobile' is added to C.
2. **Redirects** In addition to 1), articles redirected to from any w_i are added to C. For example the word 'car' redirects to 'Automobile'.
3. **Disambiguation Links** In addition to 1) and 2), all articles linked to from the disambiguation page (if any) were added to C. For example the 'Car' disambiguation page links to the 'Automobile' article, a movie and song with the title 'Cars' and several other pages.

3.1.2. Information Retrieval

The second method for identifying candidate articles makes use of an Information Retrieval system to index Wikipedia and makes use of entire articles in Wikipedia rather than just their titles. The motivation for this approach is illustrated in Table 1 which gives examples of correct mappings where the article title does not match any of the words in the synset. These mappings are found using the IR approach since the whole article content is taken into account.

Wikipedia article	WordNet synset
LIVESTOCK CARRIER	cattleship, cattle boat
FINGER COT	thumbstall
PULMONARY ALVEOLUS	alveolus, air sac, air cell
BENEFIT PERFORMANCE	benefit

Table 1: Correct mappings detected where the article title does not match any of the words in the synset.

The Terrier IR system (Ounis et al., 2007) was used to index Wikipedia with each article being treated as a document in a collection. The widely used vector space model with TF-IDF weighting (Spärck Jones, 1972) was used for retrieval. (Experiments showed that using different retrieval models did not alter the results.) Queries were formed using various terms extracted from the synset: lemmas (e.g. car, automobile), gloss (e.g. a motor vehicle with four wheels), lemmas of all related synsets (hypernyms, meronyms etc. e.g. vehicle, accelerator) and glosses of related synsets. The top ranked Wikipedia articles returned by these queries are added to C.

3.2. Selecting the best mapping

The previous stage returns a set of candidate articles for each noun synset in Wikipedia. The second stage attempt to identify the best matching article from this set using two methods: text similarity (Section 3.2.1.) and title similarity (Section 3.2.2.). The end result after applying these methods is a mapping from each noun synsets to at most one article.

3.2.1. Text Similarity

Wikipedia articles are pre-processed by removing markup then stemming and removing stopwords from the remaining text. (Other methods for pre-processing each article, including using TF-IDF weighting, did not improve performance.) Various combinations of features from the WordNet synset (lemmas, glosses, related lemmas etc.) were used. Similarity between each WordNet synset and Wikipedia article is computed using the following formula:

$$\text{text\_sim} = \frac{|A \cap B|}{min(|A|, |B|)} \quad (1)$$

where A represents the WordNet feature vector, and B represents the Wikipedia feature vector.

3.2.2. Title Similarity

The previous method use the whole Wikipedia article for comparison. However the title of the article is the single most important feature when considering similarity to the synset. Therefore a further method assigns a similarity score using the title alone. For a synset $S = \{w_1, w_2, ...w_n\}$ the $title\_score$ is computed as:

$$\text{title\_sim} = \max_{w_i \in S} \begin{cases} 1 & \text{if } title{=}w_i \\ \frac{len(title)}{len(w_i)} & \text{if substr}(title, w_i) \\ \frac{len(w_i)}{len(title)} & \text{if substr}(w_i, title) \end{cases}$$

where $len(string)$ is the length of a string and $substr(a, b)$ is true iff a is a substring of b.

3.3. Refining the mappings

The result of the mapping from WordNet to Wikipedia is a set of synset-article pairings. A global view of the mappings and information about the link structure in Wikipedia is then used to refine the mappings.

Firstly, we remove all mappings where more than one synset maps to the same Wikipedia article since these mappings are often spurious. Figure 1 shows several synsets containing the word 'tongue' that are mapped to the 'Tongue' article in Wikipedia. Only one of these synsets, with the gloss 'muscular tissue in oral cavity', represents a good match. The motivation for removing these mappings is that it is difficult for the scoring methods (in Section 3.2.) to determine the correct match, and therefore is better to simply eliminate all such cases. For the example this would mean all mappings (including the correct one) will be discarded.

The next step in refining the mappings is to exploit the links in Wikipedia to determine which of the synset-article mappings represent good matches. Figure 2 illustrates this approach. Let S denote the set of noun synsets, and A the set

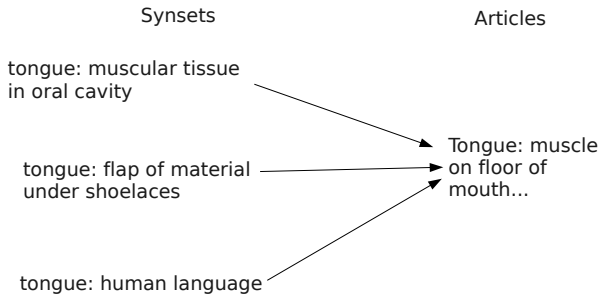

Figure 1: Multiple synsets matching a single article.

of articles which are mapped to by a synset. The articles for 'Counting', 'Accountancy' and 'Internal control' all link to, or are linked from at least one other article within the set A. This is considered as evidence that the associated synsets are good matches. However, since the article 'Exhumation' is not linked to any article in A this is excluded from the set. A further refinement is to only consider links which are reciprocal (or bidirectional). For the example case, this would mean synsets 'count' and 'accountancy' are used in the set since the associated articles link to each other. However article 'Internal control' is excluded since there are no incoming links to the associated article from within the set A.

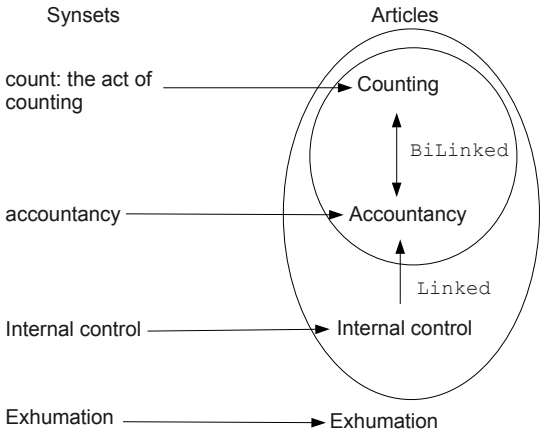

Figure 2: Links between articles

4. Annotation

In order to evaluate the mapping a set of 200 synsets, referred to as the **200NS** set, was randomly selected from WordNet and independently annotated by two annotators into one of the following five categories:

1. Matching *article*. This indicates that the article is a match for the synset, exclusively describing the same concept as the synset. If more than one article meets this requirement the best match is chosen. An example is the synset about 'poaching' (as a cooking method) with the appropriate article.

2. Related *article*. No exact matching article can be found, but a closely related one can be found. These are divided into two types:

   (a) Part-of related - The synset corresponds to part of the article, but not the whole. If more than one article meets this requirement, the most strongly related is chosen. An example is where 'tenon' is described in part of the article about 'Mortise and tenon'.

   (b) Other related - This indicates that no matching article can be found but that there is an article directly related to the synset. If more than one article meets this requirement, the most strongly related is chosen. An example is where 'bath powder' is a direct hyponym of 'Powder' as described in the article.

3. Not found. Where no article could be found, the annotators then classed the synset into one of two categories:

   (a) Dictionary term - The concept is not one that would be expected to appear in an encyclopedia. An example is found with the synset is 'dumpiness', related to the adjective for 'dumpy'. This would not be an appropriate candidate for an encyclopedic article.

   (b) Not found - The concept is one we would expect to find in an encyclopedia, but cannot be found. For example 'vegetable sheep' is a New Zealand herb but no reference could be found in Wikipedia.

The initial inter-annotator agreement was 86%, which could be considered an upper bound for automatic methods. The annotators then discussed and resolved the disagreements. The distribution of categories for the 200 articles is shown in Table 2.

Category	Synsets
1 - Match	126 (63%)
2a - Part-of related	11 (5.5%)
2b - Other related	36 (18%)
3a - Dictionary term	23 (11.5%)
3b - Not found	4 (2%)
Total	200 (100%)

Table 2: Distribution of synsets into categories.

These results show that the majority (63%) of synsets have a good matching article in Wikipedia. Additionally many synsets have articles on closely related topics, with only a few having no related article matches at all.

5. Experiments

This section evaluates the mapping methods from Section 3. Evaluation uses the 200NS set described in Section 4.

5.1. Candidate articles

The 200NS set is used to evaluate the methods for identifying candidate articles described in Section 3.1. The aim of this stage is to generate a set of articles that includes the correct mapping. A strict constraint is imposed, where only matching synset-article pairs are considered to be correct matches (not part-of or related pairs). Performance is evaluated in terms of recall, the proportion of synsets that include the correct match in the retrieved in the candidate article set.

Table 3 shows the recall generated using the title matching (Section 3.1.1.) and Information Retrieval (Section 3.1.2.) approaches. For the title matching approach results are reported using only articles (A), articles and redirects (A,R) and articles, redirects and disambiguation pages (A,R,D). Various features were used for generating queries for the Information Retrieval approach: lemmas (L), glosses (G), lemmas of related glosses (RL) and glosses of related synsets (RG).

Articles	1	5	10	20
Title matching				
A	61.1	67.5	67.5	67.5
A,R	69.1	75.4	75.4	75.4
A,R,D	**69.8**	**78.6**	**78.6**	**78.6**
Information retrieval				
L	47.6	69.1	77.8	82.5
L,G	**57.9**	**84.1**	88.1	90.5
L,RL	43.7	74.6	81.0	87.3
L,G,RL	54.8	84.1	**90.5**	**92.9**
L,G,RL,RG	34.9	65.9	73.0	78.6
Title matching & IR Combined				
A,R,L,G	74.6	**96.0**	**96.0**	**96.0**
A,R,D,L,G	73.8	92.9	92.9	93.7
A,R,L,G,RL	**74.6**	93.7	**96.0**	**96.0**
A,R,D,L,G,RL	**74.6**	92.1	93.7	93.7

Table 3: Recall (%) against number of articles combining title matching & IR methods.

For the title matching methods, adding the redirects gave a clear boost to recall performance. Using the disambiguation links also improves performance slightly. With the IR methods using the lemma, gloss and related lemmas gives the best performance, slightly better than using lemma and gloss alone.

The bottom part of Table 3 combines retrieved articles from both the title matching and IR methods. The articles from the title method are used first followed by the articles from the IR method. The best performing title matching methods (A,R and A,R,D) are combined with the best IR methods (L,G) and (L,G,RL). The results from this show that using the IR articles gives a bigger boost to recall with fewer additional articles than using the disambiguation links. This is most likely due to the fact that the disambiguation links will not be necessarily ranked in order of similarity to the synset, which is the case with the IR articles. The results from the A,R,L,G and the A,R,L,G,RL are very close, converging to the same recall performance (96.0%) after 20 articles. However the A,R,L,G reaches this level quicker, after only 10 articles.

The combined title matching and IR methods using the 'A,R,L,G' features are used to create a set of 10 candidate articles for each of the synsets in 200NS. These candidate articles are then used for the next stage in the mapping generation process.

5.2. Mapping Selection

Evaluation was performed on the 200NS set of the mappings created using the approach in Section 3.2.. Like the evaluation of the candidate selection stage (Section 5.1.), only articles that are labeled as matching articles in 200NS are considered correct mappings. For these cases the approach must identify the correct Wikipedia article to be considered correct. For other synsets, including those in 200NS labeled as related articles, the approaches must predict that there is no mapping.

Performance is computed using the following metrics. **Accuracy** is the percentage of synsets in 200NS for which the mapping is correct (by identifying either the correctly mapped Wikipedia article or that no suitable article exists). **Precision** and **recall** are computed according to the performance on finding correctly matched articles. The **F-measure** combines precision and recall measurements in the normal way, ie. $F = \frac{2PR}{P+R}$.

A similarity score is assigned to each article in the candidate set and the article with the highest score chosen as the best match using the text similarity methods described in Section 3.2. If this score exceeds a threshold the article is assigned as a positive match, otherwise it is decided there is no match for the synset. Thresholds are set using 10-fold cross validation with the J48 decision tree classifier in Weka (Hall et al., 2009). Results using different combinations of features: synset lemmas (L), synset lemmas and glosses (L,G), lemmas of the synset and related synsets (L,RL) and synset lemmas and glosses combined with lemmas of related synsets (L,G,RL). Results are averaged across the 10 folds and shown in Table 4.

Features	Acc.	Prec.	Rec.	F
Text similarity				
L	55.0	52.9	73.0	61.3
L,G	38.8	38.5	52.7	44.5
L,RL	36.5	35.9	52.4	42.6
L,G,RL	42.6	37.5	50.2	43.0
Title similarity				
Title scores	65.5	67.5	61.1	64.2
Combination				
Text similarity + title similarity	**68.4**	**75.3**	**61.2**	**67.5**

Table 4: Mapping selection performance

The best performance for the text similarity approach is achieved using lemmas alone with an accuracy of 55% and recall of 73%. Glosses and related lemmas add noise and degrade performance.

The title similarity approach (Section 3.2.) was used alone

and in combination with the best performing text similarity method (ie. using lemmas alone). The approaches were combined by taking a simple average of their scores. Results are also shown in Table 4 and show that title matching outperforms the text similarity approach. However, combining the two approaches produces better performance than either used alone. This combined approach is used to create a mapping between the synsets in 200NS and the Wikipedia articles which is referred to as the **Basic** mapping.

5.3. Mapping Refinement

The mapping refinement methods (Section 3.3.) were used to improve the mapping generated using the combined title matching and text similarity approaches. Results are shown in Table 5 where 'Basic' refers to the mapping generated using the methods in Section 5.2. 'Link-Refined' and 'BiLink-Refined' refer to the refined mappings created using the Wikipedia links and reciprocal links respectively.

Mapping	Acc.	Prec.	Rec.	F
Basic	**68.4**	75.3	**61.2**	**67.5**
Link-Refined	**68.4**	86.2	55.6	67.6
BiLink-Refined	63.9	**87.8**	46.9	61.1

Table 5: Mapping refinement performance

The precision of the predicted mapping improves when the refinement methods are applied, with the highest precision of 87.8% being achieved using the BiLinked articles. However, this increase in the accuracy of the predicted mappings is obtained at the expense of recall.

5.4. Comparison with previous approach

Direct comparison of the approach described here with alternatives ones is problematic since others have generally chosen to create a mapping in the opposite direction, i.e. from WordNet to Wikipedia. However, in order to provide some information about the effectiveness of the approach described here a comparison is carried out with the mappings generated by Ponzetto and Navigli (2010) which have been made publicly available.

One key difference between the approach presented here and the method described in (Ponzetto and Navigli, 2010) is the direction of the mapping. Given a particular article, (Ponzetto and Navigli, 2010) finds the best word sense in WordNet to match to. This means that many articles may map to a single synset. For comparison an evaluation is made here of their mapping against the 200NS dataset. Since 200NS only contains at most one article per synset, the following evaluation is applied. If any of the articles in (Ponzetto and Navigli, 2010) match the gold standard 200NS then this is marked as a true positive. All others are labelled as false matches. This means recall is artificially high, since the approach has more chances to find the right article. However precision is lower since each wrong article is considered as an incorrect match. However the figures (in parentheses) quoted in Table 6 still give some idea of the relative performance of the method. The results show that the methods described here obtain higher precision for the mappings than those obtained in (Ponzetto and Navigli, 2010).

Metric	Acc.	Prec.	Rec.	F
$ponzetto$	*(66.5)*	*(65.0)*	*(70.6)*	*(67.7)*
$ponzetto + link$	**71.0**	91.3	57.9	**70.9**
$ponzetto + bilink$	64.5	**93.6**	46.0	61.7

Table 6: Evaluation of mappings from Ponzetto and Navigli (2010) on the 200NS data. Second and third rows show effect of combining with link-refinement approach.

Further experiments show the effect of combining the link refined mappings (Section 5.3.) with the mappings of Ponzetto and Navigli (2010). Only mappings that exist in both are preserved. Using this combined approach simultaneously selects the best article for each synset and the best synset for each article. This gives the highest overall accuracy, precision and F-measure.

These results show that the approach presented here is of comparable performance to that of Ponzetto and Navigli (2010). Since the mappings are in different directions each approach provides useful information and the best results are achieved when both mappings are combined.

6. Enriching WordNet with new relations

The mapping between WordNet and Wikipedia can be used to add new relations between synsets to WordNet and thereby create an extended version of the resource which can be used for word sense disambiguation.

6.1. Deriving New Relations from Wikipedia

A mapping between all noun synsets in WordNet and Wikipedia was created using the best approaches for each stage, as determined in the previous section. Candidate articles for each synset were generated using a combination of the title matching and IR methods with the 'A,R,L,G' features (see Section 5.1.). The best mapping from each candidate set was identified using the combined 'Text similarity + title similarity' method (see Section 5.2.). In this mapping, referred to as the **Basic** mapping, 46,238 of the 82,115 noun synsets in WordNet are mapped onto a Wikipedia article and no suitable article found for the remainder. The Basic mapping was then refined to create the **Link-Refined** and **BiLink-Refined** mappings. 44,720 synsets are mapped onto a Wikipedia article in the Link-Refined mapping and 24,210 in the BiLink-Refined mapping.

These mappings can be used to derive new relations between WordNet synsets using the hyperlink structure in Wikipedia. If two synsets, a and b, are mapped onto Wikipedia articles, a' and b', and there is a hyperlink connecting a' and b' in Wikipedia then a relation between a and b is added to WordNet. For example, for the synset-article matches shown in Figure 2, new relations between 'internal control' and 'accountancy' and between 'accountancy' and 'count' would be derived from the Link-Refined

mapping and only the relation between 'accountancy' and 'count' from the BiLink-Refined mapping.
The number of relations derived from each mapping is shown in Table 7. The 'Total' column shows the total number of relations extracted from each mapping and the 'Novel' column the number of these that do not already exist in WordNet (including relations derived from the disambiguated glosses).

Mapping	Total	Novel
Basic	2,333,336	1,909,223 (81.9%)
Link-Refined	782,784	613,544 (78.4%)
BiLink-Refined	156,644	148,601 (94.9%)

Table 7: Number of relations generated from each mapping and proportion that already exist in WordNet

Table 7 shows that the many relations between WordNet synsets are derived from Wikipedia and the majority of them are novel. When the bidirectional mapping is used the number of relations identified drops to around a fifth. However, there is a larger proportion of new relations compared to using directional links. This might be due to the fact that most existing relations in WordNet are between hypernyms and hyponyms, which are directional relations, or an indication that more of the bi-directional relations are topically related or co-ordinate terms.

6.2. Word Sense Disambiguation

The new relations are evaluated on a Word Sense Disambiguation (WSD) task. The UKB system (Agirre and Soroa, 2009) is used as the WSD system. This represents a lexical knowledge base, such as WordNet, as a graph. This graph is created by representing each synset as a vertex and adding edges between them if they are related in WordNet. Both the relations encoded in WordNet (hypernyms, meronyms etc.) and those that can be derived from the disambiguated glosses are used to add edges to the graph. To enrich WordNet with the relations derived from Wikipedia new edges are simply added to the graph. The UKB system applies the Personalized PageRank to rank the vertices and thus perform disambiguation. The more accurate `ppr_w2w` algorithm which builds a separate graph for each target word in context is used.
The SemEval 2007 coarse grained all words task (Navigli et al., 2007) is used for evaluation. Experiments are carried out using the nouns in this data set. The accuracy of the WSD system is computed as the percentage of tokens that are correctly disambiguated.
Results of the WSD evaluation using relations derived from the mappings described in the previous section are shown in Table 8. The best performance is obtained when the relations from the BiLink-Refined mapping are added. These results demonstrate that WSD performance improves with the addition of the smaller set of more accurate relations in the WN3+BiLink-Refined set compared to those containing a greater number of (less accurate) relations. However there is no significant improvement over the WN3 baseline. Table 9 compares the best result with recent state of the art approaches on the SemEval 2007 task. These are the best

Method	Accuracy
WN3	84.0
WN3+Basic	80.4
WN3+LinkRefined	83.3
WN3+BiLink-Refined	**84.1**

Table 8: WSD performance on SemEval 2007 coarse grained all words task.

performing unsupervised system in SemEval 2007 (Koeling and McCarthy, 2007), the best supervised system (Chan et al., 2007), and a knowledge-rich system (Navigli and Velardi, 2005) which participated outside the competition.

Method	Accuracy
WN3 + BiLink-Refined	**84.1**
(Koeling and McCarthy, 2007)	81.1
(Chan et al., 2007)	82.3
(Navigli and Velardi, 2005)	**84.1**

Table 9: Comparison with state of the art.

Performance of the WSD system using the enriched WordNet is comparable with the state of the art SSI system (Navigli and Velardi, 2005) and outperforms the best supervised and unsupervised entries to SemEval 2007. It should be noted that all these systems use the most frequent sense as a backoff strategy when no sense can be identified whereas our approach is completely unsupervised and does not use any information about the frequency of senses. It is possible that using such information could boost performance further.

7. Conclusions

This paper describes a mapping approach from WordNet synsets to Wikipedia articles. For evaluation purposes set of synsets has been manually annotated with associated Wikipedia articles. This gives an analysis of the overlap between noun synsets and Wikipedia articles, with over 60% of the synsets having a good matching article.
The first stage of the mapping process reduces the search space from 3 million to less than 20 articles, while preserving recall at 96%. The subsequent stages achieve 87.8% precision and 46.9% recall using global refinement approaches. The full mappings are made available online[1].

8. Acknowledgements

The research leading to these results was supported by the PATHS project (`http://paths-project.eu`) funded by the European Communitys Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 270082.

9. References

E. Agirre and A. Soroa. 2009. Personalizing PageRank for Word Sense Disambiguation. In *Proceedings of the 12th*

[1] `http://staffwww.dcs.shef.ac.uk/people/S.Fernando`

Conference of the European Chapter of the Association for Computational Linguistics, pages 33–41. Association for Computational Linguistics.

R. Benassi, S. Bergamaschi, and M. Vincini. 2004. Web Semantic Search with TUCUXI. In *Proceedings of the Twelfth Italian Symposium on Advanced Database Systems SEBD*, pages 426–433.

G. Carenini, Raymond T. Ng, and X. Zhou. 2008. Summarizing Emails with Conversational Cohesion and Subjectivity. In *Proceedings of the Association for Computational Linguistics 2008: Human Language Technology*, pages 353–361, Columbus, Ohio, June. Association for Computational Linguistics.

Y.S. Chan, H.T. Ng, and Z. Zhong. 2007. NUS-PT: Exploiting parallel texts for Word Sense Disambiguation in the English all-words tasks. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 253–256. Association for Computational Linguistics.

M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I.H. Witten. 2009. The WEKA data mining software: An update. *Association for Computing Machinery Special Interest Group on Knowledge Discovery and Data Mining Explorations Newsletter*, 11(1):10–18.

R. Koeling and D. McCarthy. 2007. Sussx: WSD using automatically acquired predominant senses. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 314–317. Association for Computational Linguistics.

R. Navigli and P. Velardi. 2005. Structural semantic interconnections: a knowledge-based approach to Word Sense Disambiguation. *Institute of Electrical and Electronics Engineers Transactions on Pattern Analysis and Machine Intelligence*, pages 1075–1086.

R. Navigli, K.C. Litkowski, and O. Hargraves. 2007. Semeval-2007 task 07: Coarse-grained English all-words task. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 30–35. Association for Computational Linguistics.

I. Ounis, C. Lioma, C. Macdonald, and V. Plachouras. 2007. Research directions in terrier. *Novatica/UPGRADE Special Issue on Web Information Access*.

S. P. Ponzetto and R. Navigli. 2010. Knowledge-rich Word Sense Disambiguation rivaling supervised system. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics,* Uppsala, Sweden, 11–16 July 2010, pages 1522–1531.

M. Ruiz-Casado, E. Alfonseca, and P. Castells. 2005. Automatic Assignment of Wikipedia Encyclopedic Entries to WordNet Synsets. *Advances in Web Intelligence: Third International Atlantic Web Intelligence Conference*.

K. Spärck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21.

F. M. Suchanek, G. Kasneci, and G. Weikum. 2008. YAGO: A Large Ontology from Wikipedia and WordNet. *Journal of Web Semantics*.