RACAI: Unsupervised WSD experiments @ SemEval-2, Task #17

Radu Ion
Institute for AI, Romanian Academy
13, Calea 13 Septembrie, Bucharest
050711, Romania
radu@racai.ro

Dan Ştefănescu
Institute for AI, Romanian Academy
13, Calea 13 Septembrie, Bucharest
050711, Romania
danstef@racai.ro

Abstract

This paper documents the participation of the Research Institute for Artificial Intelligence of the Romanian Academy (RACAI) to the Task 17 – All-words Word Sense Disambiguation on a Specific Domain, of the SemEval-2 competition. We describe three unsupervised WSD systems that make extensive use of the Princeton WordNet (WN) structure and WordNet Domains in order to perform the disambiguation. The best of them has been ranked the 12th by the task organizers out of 29 judged runs.

1 Introduction

Referring to the last SemEval (SemEval-1, (Agirre et al., 2007a)) and to our recent work (Ion and Ştefănescu, 2009), unsupervised Word Sense Disambiguation (WSD) is still at the bottom of WSD systems ranking with a significant loss in performance when compared to supervised approaches. With Task #17 @ SemEval-2, this observation is (probably[1]) reinforced but another issue is re-brought to light: the difficulty of supervised WSD systems to adapt to a given domain (Agirre et al., 2009). With general scores lower with at least 3% than 3 years ago in Task #17 @ SemEval-1 which was a supposedly harder task (general, no particular domain WSD was required for all words), we observe that supervised WSD is certainly more difficult to implement in a real world application.

Our unsupervised WSD approach benefited from the specification of this year's Task #17 which was a domain-limited WSD, meaning that the disambiguation would be applied to content words drawn from a specific domain: the surrounding environment. We worked under the assumption that a term of the given domain would have the same meaning with all its occurrences throughout the text. This hypothesis has been put forth by Yarowsky (1993) as the "one sense per discourse" hypothesis (OSPD for short).

The task organizers offered a set of background documents with no sense annotations to the competitors who want to train/tune their systems using data from the same domain as the official test set. Working with the OSPD hypothesis, we set off to construct/test domain specific WSD models from/on this corpus using the WordNet Domains (Bentivogli et al., 2004). For testing purposes, we have constructed an in-house gold standard from this corpus that comprises of 1601 occurrences of 204 terms of the "surrounding environment" domain that have been automatically extracted with the highest confidence. We have observed that our gold standard (which has been independently annotated by 3 annotators but on non-overlapping sections which led to having no inter-annotator agreement scores) obeys the OSPD hypothesis which we think that is appropriate to domain-limited WSD.

In what follows, we will briefly acknowledge the usage of WordNet Domains in WSD, we will then describe the construction of the corpus of the background documents including here the creation of an in-house gold standard, we will then briefly describe our three WSD algorithms and finally we will conclude with a discussion on the ranking of our runs among the 29 evaluated by the task organizers.

2 Related Work

WordNet Domains is a hierarchy of labels that have been assigned to WN synsets in a one to (possible) many relationship (but the frequent case is a single WN domain for a synset). A domain is the name of an area of knowledge that is recognized as unitary (Bentivogli et al., 2004).

[1] At the time of the writing we only know the systems ranking without the supervised/unsupervised distinction.

Proceedings of the 5th International Workshop on Semantic Evaluation, ACL 2010, pages 411–416,
Uppsala, Sweden, 15-16 July 2010. ©2010 Association for Computational Linguistics

Thus labels such as “*architecture*”, “*sport*” or “*medicine*” are mapped onto synsets like “*arch(4)*-noun”, “*playing(2)*-noun” or “*chronic(1)*-adjective” because of the fact that the respective concept evokes the domain.

WordNet Domains have been used in various ways to perform WSD. The main usage of this mapping is that the domains naturally create a clustering of the WN senses of a literal thus offering a sense inventory that is much coarser than the fine sense distinctions of WN. For instance, senses 1 (“*a flat-bottomed motor vehicle that can travel on land or water*”) and 2 (“*an airplane designed to take off and land on water*”) of the noun “*amphibian*” are both mapped to the domain “*transport*” but the 3rd sense of the same noun is mapped onto the domains “*animals/biology*” being the “*cold-blooded vertebrate typically living on land but breeding in water; aquatic larvae undergo metamorphosis into adult form*” (definitions from version 2.0 of the WN).

Vázquez et al. (2004) use WordNet Domains to derive a new resource they call the Relevant Domains in which, using WordNet glosses, they extract the most representative words for a given domain. Thus, for a word w and a domain d, the Association Ratio formula between w and d is

$$\mathrm{AR}(w,d) = \mathrm{P}(w\,|\,d)\cdot\log_2\frac{\mathrm{P}(w\,|\,d)}{\mathrm{P}(w)}$$

in which, for each synset its gloss has been POS tagged and lemmatized. The probabilities are computed counting pairs $\langle w,d\rangle$ in glosses (each gloss has an associated d domain via its synset).

Using the Relevant Domains, the WSD procedure for a given word w in its context C (a 100 words window centered in w), computes a similarity measure between two vectors of AR scores: the first vector is the vector of AR scores of the sentence in which w appears and the other is the vector of domain scores computed for the gloss of a sense of w (both vectors are normalized such that they contain the same domains). The highest similarity gives the sense of w that is closest to the domain vector of C. With this method, Vázquez et al. obtain a precision of 0.54 and a recall of 0.43 at the SensEval-2, English All-Words Task placing them in the 10th position out of 22 systems where the best one (a supervised system) achieved a 0.69 precision and an equal recall.

Another approach to WSD using the WordNet Domains is that of Magnini et al. (2002). The method is remarkably similar to the previous one in that the description of the vectors and the selection of the assigned sense is the same. What differs, is the weights that are assigned to each domain in the vector. Magnini et al. distinguish between text vectors (C vectors in the previous presentation) and sense vectors. Text (or context) vector weights are computed comparing domain frequency in the context with the domain frequency over the entire corpus (see Magnini et al. (2002) for details). Sense vectors are derived from sense-annotated data which qualifies this method as a supervised one. The results that have been reported at the same task the previous algorithm participated (SensEval-2, English All-Words Task), are: precision 0.748 and recall 0.357 (12th place).

Both the methods presented here are very simple and easy to adapt to different domains. One of our methods (RACAI-1, see below) is even simpler (because it makes the OSPD simplifying assumption) and performs with approximately the same accuracy as any of these methods judging by the rank of the system and the total number of participants.

3 Using the Background Documents collection

Task #17 organizers have offered a set of background documents for training/tuning/testing purposes. The corpus consists of 124 files from the “surrounding environment” domain that have been collected in the framework of the Kyoto Project (http://www.kyoto-project.eu/).

First, we have assembled the files into a single corpus in order to be able to apply some cleaning procedures. These procedures involved the removal of the paragraphs in which the proportion of letters (Perl character class “`[A-Za-z_-]`”) was less than 0.8 because the text contained a lot of noise in form of lines of numbers and other symbols which probably belonged to tables. The next stage was to have the corpus POS-tagged, lemmatized and chunked using the TTL web service (Tufiş et al., 2008). The resulting file is an XML encoded corpus which contains 136456 sentences with 2654446 tokens out of which 348896 are punctuation tokens.

In order to test our domain constrained WSD algorithms, we decided to construct a test set with the same dimension as the official test set of about 2000 occurrences of content words specific to the “surrounding environment” domain. In doing this, we have employed a simple term extraction algorithm which considers that terms, as opposed to words that are not domain specific, are not evenly distributed throughout the corpus. To formalize this, the corpus is a vector of lemmas $C = [l_1, l_2, \ldots, l_N]$ and for each unique lemma $l_j, 1 \le j \le N$, we compute the mean of the absolute differences of its indexes in C as

$$\mu = \frac{\sum\limits_{1 \le j < k \le N} |j - k|}{f(l_j) - 1}, l_j = l_k \wedge \forall m, j < m < k, l_j \ne l_m$$

where $f(l_j)$ is the frequency of l_j in C. We also compute the standard deviation of these differences from the mean as

$$\sigma = \sqrt{\frac{\sum\limits_{1 \le j < k \le N} (|j - k| - \mu)^2}{f(l_j) - 2}}$$

in the same conditions as above.

With the mean and standard deviation of indexes differences of a content word lemma computed, we construct a list of all content word lemmas that is sorted in descending order by the quantity σ / μ which we take as a measure of the evenness of a content word lemma distribution. Thus, lemmas that are in the top of this list are likely to be terms of the domain of the corpus (in our case, the "surrounding environment" domain). Table 1 contains the first 20 automatically extracted terms along with their term score.

Having the list of terms of our domain, we have selected the first *ambiguous* 210 (which have more than 1 sense in WN) and constructed a test set in which each term has (at least) 10 occurrences in order to obtain a test corpus with at least 2000 occurrences of the terms of the "surrounding environment" domain. A large part of these occurrences have been independently sense-annotated by 3 annotators which worked on disjoint sets of terms (70 terms each) in order to finish as soon as possible. In the end we managed to annotate 1601 occurrences corresponding to 204 terms.

When the gold standard for the test set was ready, we checked to see if the OSPD hypothesis holds. In order to determine if it does, we computed the average number of annotated different senses per term which is 1.36. In addition, considering the fact that out of 204 annotated terms, 145 are annotated with a single sense, we may state that in this case, the OSPD hypothesis holds.

Term	**Score**	**Term**	**Score**
gibbon	15.89	Oceanica	9.41
fleet	13.91	orangutan	9.19
sub-region	13.01	laurel	9.08
Amazon	12.41	coral	9.06
roundwood	12.26	polar	9.05
biocapacity	12.23	wrasse	8.80
footprint	11.68	reef	8.78
deen	11.45	snapper	8.67
dune	10.57	biofuel	8.53
grouper	9.67	vessel	8.35

Table 1: The first 20 automatically extracted terms of the "surrounding environment" domain

4 The Description of the Systems

Since we are committed to assign a unique sense per word in the test set, we might as well try to automatically induce a *WSD model* from the background corpus in which, for each lemma along with its POS tag that also exists in WN, a single sense is listed that is derived from the corpus. Then, for any test set of the same domain, the algorithm would give the sense from the WSD model to any of the occurrences of the lemma.

What we actually did, was to find a list of most frequent 2 WN domains (frequency count extracted from the *whole corpus*) for each lemma with its POS tag, and using these, to list all senses of the lemma that are mapped onto these 2 domains (thus obtaining a reduction of the average number of senses per word). The steps of the algorithm for the creation of the WSD model are:

1. in the given corpus, for each lemma *l* and its POS-tag *p* normalized to WN POS notation ("n" for nouns, "v" for verbs, "a" for adjectives and "b" for adverbs), for each of its senses from WN, increase by 1 each frequency of each mapped domain;
2. for each lemma *l* with its POS-tag *p*, retain only those senses that map onto the most frequent 2 domains as determined by the frequency list from the first step.

Using our 2.65M words background corpus to build such a model (Table 2 contains a sample), we have obtained a decrease in average ambiguity degree (the average number of senses per content word lemma) from 2.43 to 2.14. If we set a threshold of at least 1 for the term score of the lemmas to be included into the WSD model (which selects 12062 lemmas, meaning about 1/3 of all unique lemmas in the corpus), we obtain the same reduction thus contradicting our hypothesis that the average ambiguity degree of terms would be reduced more than the average ambiguity degree of all words in the corpus. This result might be due to the fact that the "*factotum*" domain is very frequent (much more frequent than any of the other domains).

Lemma	POS:Total no. of WN senses	First 2 selected domains	Selected senses
fish	n:2	animals,biology	1
Arctic	n:1	geography	1
coral	n:4	chemistry,animals	2,3,4

Table 2: A sample of the WSD model built from the background corpus

In what follows, we will present our 3 systems that use WSD models derived from the test sets (both the in-house and the official ones). In the Results section we will explain this choice.

4.1 RACAI-1: WordNet Domains-driven, Most Frequent Sense

The first system, as its name suggests, is very simple: using the WSD model, it chooses the most frequent sense (MFS) of the lemma *l* with POS *p* according to WN (that is, the lowest numbered sense from the list of senses the lemma has in the WSD model).

Trying this method on our in-house developed test set, we obtain encouraging results: the overall accuracy (precision is equal with the recall because all test set occurrences are tried) is at least 4% over the general MFS baseline (sense no. 1 in all cases). The Results section gives details.

4.2 RACAI-2: The Lexical Chains Selection

With this system, we have tried to select only one sense (not necessarily the most frequent one) of lemma *l* with POS *p* from the WSD model. The selection procedure is based on lexical chains computation between senses of the target word (the word to be disambiguated) and the content words in its sentence in a manner that will be explained below.

We have used the lexical chains description and computation method described in (Ion and Ştefănescu, 2009). To reiterate, a lexical chain is not simply a set of topically related words but becomes a path of synsets in the WordNet hierarchy. The lexical chain procedure is a function of two WN synsets, LXC(s_1, s_2), that returns a semantic relation path that one can follow to reach s_2 from s_1. On the path from s_2 to s_1 there are *k* synsets ($k \geq 0$) and between 2 adjacent synsets there is a WN semantic relation. Each lexical chain can be assigned a certain score that we interpret as a measure of the semantic similarity (SS) between s_1 and s_2 (see (Ion and Ştefănescu, 2009) and (Moldovan and Novischi, 2002) for more details). Thus, the higher the value of SS(s_1, s_2), the higher the semantic similarity between s_1 and s_2.

We have observed that using RACAI-1 on our in-house test set but allowing it to select the most frequent **2** senses of lemma *l* with POS *p* from the WSD model, we obtain a whopping **82% accuracy**. With this observation, we tried to program RACAI-2 to make a binary selection from the first 2 most frequent senses of lemma *l* with POS *p* from the WSD model in order to approach the 82% percent accuracy limit which would have been a very good result. The algorithm is as follows: for a lemma *l* with POS *p* and a lemma l_c with POS p_c from the context (sentence) of *l*, compute the best lexical chain between any of the first 2 senses of *l* and any of the first 2 senses of l_c according to the WSD model. If the first 2 senses of *l* are *a* and *b* and the first 2 senses of l_c are *x* and *y* and the best lexical chain score has been found between *a* and *y* for instance, then credit sense *a* of *l* with SS(*a*, *y*). Sum over all l_c from the context of *l* and select that sense of *l* which has a maximum semantic similarity with the context.

4.3 RACAI-3: Interpretation-based Sense Assignment

This system tries to generate all the possible sense assignments (called interpretations) to the lemmas in a sentence. Thus, in principle, for each content word lemma, all its WN senses are considered thus generating an exponential explosion of the sense assignments that can be attributed to a sentence. If we have *N* content word lemmas which have *k* senses on average, we obtain a search space of k^N interpretations which have to be scored.

Using the observation mentioned above that the first 2 senses of a lemma according to the WSD model yields a performance of 82%, brings the search space to 2^N but for a large *N*, it is still too big.

The solution we adopted (besides considering the first 2 senses from the WSD model) consists in segmenting the input sentence in *M* independent segments of 10 content word lemmas each, which will be processed independently, yielding a search space of at most $M \cdot 2^{10}$ of smaller interpretations. The best interpretation per each segment would thus be a part of the best interpretation of the sentence. Next, we describe how we score an interpretation.

For each sense *s* of a lemma *l* with POS *p* (from the first 2 senses of *l* listed in the WSD model) we compute an associated set of content words (lemmas) from the following sources:

- all content word lemmas extracted from the sense *s* corresponding gloss (disregarding the auxiliary verbs);
- all literals of the synset in which lemma *l* with sense *s* exists;
- all literals of the synsets that are linked with the synset *l*(*s*) by a relation of the following type: *hypernym*, *near_antonym*, *eng_derivative*, *hyponym*, *meronym*, *holonym*, *similar_to*, *derived*;
- all content word lemmas extracted from the glosses corresponding to synsets that are linked with the *l*(*s*) synset by a relation of the following type: *hypernym*, *eng_derivative*, *similar_to*, *derived*;

With this feature set V of a sense *s* belonging to lemma *l* with POS *p*, for a given interpretation (a specific assignment of senses to each lemma in a segment), its score S (initially 0) is computed iteratively (for two adjacent position *i* and *i* + 1 in the segment) as

$$\mathrm{S} \leftarrow \mathrm{S} + |\mathrm{V}_i \cap \mathrm{V}_{i+1}|, \quad \mathrm{V}_{i+1} \leftarrow \mathrm{V}_i \cup \mathrm{V}_{i+1}$$

where the |X| function is the cardinality function on the set X and ← is the assignment operator.

5 Results

In order to run our WSD algorithms, we had to extract WSD models. We tested the accuracy of the disambiguation (onto the in-house developed gold standard) with RACAI-1 and RACAI-2 systems (RACAI-3 was not ready at that time) with models extracted **a)** from the whole background corpus and **b)** from the in-house developed test set (named here the RACAI test set, see section 3). The results are reported in Table 3 along with RACAI-1 system returning the first 2 senses of a lemma from the WSD model and the general MFS baseline.

As we can see, the results with the WSD model extracted from the test set are marginally better than the other results. This was the reason for which we chose to extract the WSD model from the official test set as opposed to using the WSD model extracted from the background corpus.

	RACAI Test Set	**Background Corpus**
RACAI-1	0.647	0.644
RACAI-1 (2 senses)	0.825	0.811
RACAI-2	0.591	0.582
MFS (sense no. 1)	0.602	0.602

Table 3: RACAI systems results (accuracy) on the RACAI test set

However, we did not research the possibility of adding the official test set to either the RACAI test set or the background corpus and extract WSD models from there.

The official test set (named the SEMEVAL test set here) contains 1398 occurrences of content words for disambiguation, out of which 366 are occurrences of verbs and 1032 are occurrences of nouns. These occurrences correspond to 428 lemmas. Inspecting these lemmas, we have found that there are many of them which are not domain specific (in our case, specific to the "surrounding environment" domain). For instance, the verb to "*be*" is at the top of the list with 99 occurrences. It is followed by the noun "*index*" with 32 occurrences and by the noun "*network*" with 22 occurrences. With fewer occurrences follow "*use*", "*include*", "*show*", "*provide*", "*part*" and so on. Of course, the SEMEVAL test set includes proper terms of the designated domain such as "*area*" (61 occurrences), "*species*" (58 occurrences), "*nature*" (31 occurrences), "*ocean*", "*sea*", "*water*", "*planet*", etc.

Table 4 lists our official results on the SEMEVAL test set.

	Precision	**Recall**	**Rank**
RACAI-1	0.461	0.46	#12
RACAI-2	0.351	0.35	#25
RACAI-3	0.433	0.431	#18
MFS	0.505	0.505	#6

Table 4: RACAI systems results (accuracy) on the SEMEVAL test set

Precision is not equal to recall because of the fact that our POS tagger found two occurrences of the verb to "*be*" as auxiliaries which were ignored. The column Rank indicates the place our systems have in a 29 run ranking of all systems that participated in Task 17 – All-words Word Sense Disambiguation on a Specific Domain, of the SemEval-2 competition which was won by a system that achieved a precision of 0.57 and a recall of 0.555.

The differences with the runs on the RACAI test set are significant but this can be explained by the fact that our WordNet Domains WSD method cannot cope with general (domain independent) WSD requirements in which the "one sense per discourse" hypothesis does not necessarily hold.

6 Conclusions

Regarding the 3 systems that we entered in the Task #17 @ SemEval-2, we think that the lexical chains algorithm (RACAI-2) is the most promising even if it scored the lowest of the three. We attribute its poor performances to the lexical chains computation, especially to the weights of the WN semantic relations that make up a chain. Also, we will extend our research regarding the correctness of lexical chains (the degree to which a human judge will appreciate as correct or evocative or as common knowledge a semantic path between two synsets).

We also want to check if our three systems make the same mistakes or not in order to devise a way in which we can combine their outputs.

RACAI is at the second participation in the SemEval series of WSD competitions. We are committed to improving the unsupervised WSD technology which, we think, is more easily adaptable and usable in real world applications. We hope that SemEval-3 will reveal significant improvements in this direction.

Acknowledgments

The work reported here was supported by the Romanian Ministry of Education and Research through the STAR project (no. 742/19.01.2009).

References

Eneko Agirre, Lluís Màrquez and Richard Wicentowski, Eds., 2007. *Proceedings of Semeval-2007 Workshop*. Prague, Czech Republic: Association for Computational Linguistics, 2007.

Eneko Agirre, Oier Lopez de Lacalle, Christiane Fellbaum, Andrea Marchetti, Antonio Toral, Piek Vossen. 2009. *SemEval-2010 Task 17: All-words Word Sense Disambiguation on a Specific Domain*. In Proceedings of NAACL workshop on Semantic Evaluations (SEW-2009). Boulder,Colorado, 2009.

Luisa Bentivogli, Pamela Forner, Bernardo Magnini and Emanuele Pianta. 2004. *Revising WordNet Domains Hierarchy: Semantics, Coverage, and Balancing*. In COLING 2004 Workshop on "Multilingual Linguistic Resources", Geneva, Switzerland, August 28, 2004, pp. 101-108.

Radu Ion and Dan Ştefănescu. 2009. Unsupervised Word Sense Disambiguation with Lexical Chains and Graph-based Context Formalization. In Zygmunt Vetulani, editor, Proceedings of the 4th Language and Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics, pages 190–194, Poznań, Poland, November 6–8 2009. Wydawnictwo Poznańskie Sp.

Bernardo Magnini, Carlo Strapparava, Giovanni Pezzulo, Alfio Gliozzo. 2002. *The role of domain information in Word Sense Disambiguation*. Natural Language Engineering, 8(4), 359—373, December 2002.

Dan Moldovan and Adrian Novischi. 2002. *Lexical chains for question answering*. In Proceedings of the 19th International Conference on Computational Linguistics, August 24 – September 01, 2002, Taipei, Taiwan, pp. 1—7.

Dan Tufiş, Radu Ion, Alexandru Ceauşu and Dan Ştefănescu. 2008. *RACAI's Linguistic Web Services*. In Proceedings of the 6th Language Resources and Evaluation Conference – LREC 2008, Marrakech, Morocco, May 2008. ELRA – European Language Ressources Association. ISBN 2-9517408-4-0.

Sonia Vázquez, Andrés Montoyo and German Rigau. 2004. *Using Relevant Domains Resource for Word Sense Disambiguation*. In Proceedings of the International Conference on Artificial Intelligence (IC-AI'04), Las Vegas, Nevada, 2004.

David Yarowsky. 1993. *One sense per collocation*. In ARPA Human Language Technology Workshop, pp. 266–271, Princeton, NJ, 1993.