Improvements to Monolingual English Word Sense Disambiguation*

Weiwei Guo
Computer Science Department
Columbia University
New York, NY, 10115, USA
wg2162@cs.columbia.edu

Mona T. Diab
Center for Computational Learning Systems
Columbia University
New York, NY 10115, USA
mdiab@ccls.columbia.edu

Abstract

Word Sense Disambiguation remains one of the most complex problems facing computational linguists to date. In this paper we present modification to the graph based state of the art algorithm In-Degree. Our modifications entail augmenting the basic Lesk similarity measure with more relations based on the structure of WordNet, adding SemCor examples to the basic WordNet lexical resource and finally instead of using the LCH similarity measure for computing verb verb similarity in the In-Degree algorithm, we use JCN. We report results on three standard data sets using three different versions of WordNet. We report the highest performing monolingual unsupervised results to date on the Senseval 2 all words data set. Our system yields a performance of 62.7% using WordNet 1.7.1.

1 Introduction

Despite the advances in natural language processing (NLP), Word Sense Disambiguation (WSD) is still considered one of the most challenging problems in the field. Ever since the field's inception, WSD has been perceived as one of the central problems in NLP as an enabling technology that could potentially have far reaching impact on NLP applications in general. We are starting to see the beginnings of a positive effect of WSD in NLP applications such as Machine Translation (Carpuat and Wu, 2007; Chan et al., 2007). Advances in research on WSD in the current millennium can be attributed to several key factors: the availability of large scale computational lexical resources such as WordNets (Fellbaum, 1998; Miller, 1990), the availability of large scale corpora, the existence and dissemination of standardized data sets over the past 10 years through the different test beds of SENSEVAL and SEMEVAL competitions,[1] devising more robust computing algorithms to handle large scale data sets, and simply advancement in hardware machinery.

In this paper, we address the problem of WSD of all the content words in a sentence. In this framework, the task is to associate all tokens with their contextually relevant meaning definitions from some computational lexical resource. We present an enhancement on an existing graph based algorithm, In-Degree, as described in (Sinha and Mihalcea, 2007). Like the previous work, our algorithm is unsupervised. We show significant improvements over previous state of the art performance on several existing data sets, SENSEVAL2, SENSEVAL3 and SEMEVAL.

2 Word Sense Disambiguation

The definition of WSD has taken on several different meanings in recent years. In the latest SEMEVAL (2007) workshop, there were 18 tasks defined, several of which were on different languages, however we notably recognize the widening of the definition of the task of WSD. In addition to the traditional all words and lexical sample tasks, we note new tasks on word sense discrimination (no sense inventory is needed, the different senses are merely distinguished), lexical substitution using synonyms of words as substitutes, as well as meaning definitions obtained from different languages namely using words in translation.

Our paper is about the classical all words task of WSD. In this task, all the content bearing words in a running text are disambiguated from a static lexical

*The second author has been partially funded by DARPA GALE project. We would also like to thank the useful comments rendered by three anonymous reviewers.

[1] http://www.semeval.org

Proceedings of the NAACL HLT Workshop on Semantic Evaluations: Recent Achievements and Future Directions, pages 64–69,
Boulder, Colorado, June 2009. ©2009 Association for Computational Linguistics

resource. For example a sentence such as *I walked by the bank and saw many beautiful plants there.* will have the verbs *walked, saw*, the nouns *bank, plants,* the adjectives *many, beautiful,* and the adverb *there*, be disambiguated from a standard lexical resource. Hence using WordNet,[2] *walked* will be assigned the meaning to *u*se one's feet to advance; advance by steps, *saw* will be assigned the meaning to *p*erceive by sight or have the power to perceive by sight, the noun *bank* will be assigned the meaning *s*loping land especially the slope beside a body of water and so on.

3 Related Works

Many systems over the years have been used for the task. A thorough review of the current state of the art is in (Navigli, 2009). Several techniques have been used to tackle the problem ranging from rule based/knowledge based approaches to unsupervised and supervised machine learning approaches. To date, the best approaches that solve the all words WSD task are supervised as illustrated in the different SenseEval and SEMEVAL All Words tasks (M. Palmer and Dang, 2001; Snyder and Palmer, 2004; Pradhan et al., 2007).

In this paper, we present an unsupervised approach to the all words WSD problem relying on WordNet similarity measures. We will review only three of the most relevant related research due to space limitations. We acknowledge the existence of many research papers that tackled the problem using unsupervised approaches.

Firstly, in work by (Pedersen and Patwardhan, 2005), the authors investigate different word similarity measures as a means of disambiguating words in context. They compare among different similarity measures. They show that using an extension on the Lesk similarity measure (Lesk, 1986) between the target words and their contexts and the contexts of those of the WordNet synset entries (Gloss Overlap), outperforms all the other similarity measures. Their approach is unsupervised. They exploit the different relations in WordNet. They also go beyond the single word overlap, they calculate the overlap in n-grams. They report results on the English Lexical sample task from Senseval 2 which comprised nouns, verbs and adjectives. The majority of the words in this set is polysemous. They achieve an F-measure of 41.2% on nouns, 21.2% on verbs, and 25.1% on adjectives.

The second related work to ours is the work by (Mihalcea, 2005). Mihalcea (2005) introduced a graph based unsupervised technique for all word sense disambiguation. Similar to the previous study, the author relied on the similarity of the WordNet entry glosses using the Lesk similarity measure. The study introduces a graph based sequence model of the problem. All the open class words in a sentence are linked via an undirected graph where all the possible senses are listed. Then dependency links are drawn between all the sense pairs. Weights on the arcs are determined based on the semantic similarity using the Lesk measure. The algorithm is basically to walk the graph and find the links with the highest possible weights deciding on the appropriate sense for the target words in question. This algorithm yields an overall F-score of 54.2% on the Senseval 2 all words data set and an F-score of 64.2% on nouns alone.

Finally, the closest study relevant to the current paper yields state of the art performance is an unsupervised approach described in (Sinha and Mihalcea, 2007). In this work, the authors combine different semantic similarity measures with different graph based algorithms as an extension to work in (Mihalcea, 2005). The authors proposed a graph-based WSD algorithm. Given a sequence of words $W = \{w_1, w_2...w_n\}$, each word w_i with several senses $\{s_{i1}, s_{i2}...s_{im}\}$. A graph G = (V,E) is defined such that there exists a vertex v for each sense. Two senses of two different words may be connected by an edge e, depending on their distance. That two senses are connected suggests they should have influence on each other, so normally a maximum allowable distance is set. They explore 4 different graph based algorithms. The highest yielding algorithm in their work is the `In-Degree` algorithm combining different WordNet similarity measures depending on POS. They used the Jiang and Conrath (JCN) (Jiang and Conrath., 1997) similarity measure within nouns, the Leacock & Chodorow (LCH) (Leacock and Chodorow, 1998) similarity measure within verbs, and the Lesk (Lesk, 1986) similarity measure within adjectives and within adverbs and

[2] http://wordnet.princeton.edu

across different POS tags. They evaluate their work against the Senseval 2 all words task. They tune the parameters of their algorithm – specifically the normalization ratio for some of these measures — based on the Senseval 3 data set. They report a state of the art unsupervised system that yields an overall performance of 57.2%.

4 Our Approach

In this paper, we extend the (Sinha and Mihalcea, 2007) work (hence forth SM07) in some interesting ways. We focus on the `In-Degree` graph based algorithm as it was the best performer in the SM07 work. The *In-Degree* algorithm presents the problem as a weighted graph with senses as nodes and similarity between senses as weights on edges. The In-Degree of a vertex refers to the number of edges incident on that vertex. In the weighted graph, the In-Degree for each vertex is calculated by summing the weights on the edges that are incident on it.

After all the In-Degree values for each sense is computed, the sense with maximum value is chosen as the final sense for that word. SM07 combine different similarity measures. They show that best combination is JCN for noun pairs and LCH for verb pairs, and Lesk for within adjectives and within adverbs and also across different POS, for example comparing senses of verbs and nouns. Since different similarity measures use different similarity scales, SM07 did not directly use the value returned from the similarity metrics. Instead, the values were normalized. Lesk value is observed in a range from 0 to an arbitrary value, so values larger than 240 were set to 1, and the rest is mapped to an interval [0,1]. Similarily JCN and LCH were normalized to the interval from [0,1].[3]

In this paper, we use the basic In-Degree algorithm while applying some modifications to the basic similarity measures exploited and the WordNet lexical resource. Similar to the original In-Degree algorithm, we produce a probabilistic ranked list of senses. Our modifications are described as follows:

JCN for Verb-Verb Similarity In our implementation of the In-Degree algorithm, we use the JCN similarity measure for both Noun-Noun similarity calculation similar to SM07. In addition, instead of using LCH for Verb-Verb similarity, we use JCN for Verb Verb similarity based on our empirical observation on SENSEVAL 3 data, JCN yields better performance than when employing LCH among verbs.

Expand Lesk Following the intuition in (Pedersen and Patwardhan, 2005) – henceforth (PEA05) – we expand the basic Lesk similarity measure to take into account the glosses for all the relations for the synsets on the contextual words and compare them with the glosses of the target word senses, hence going beyond the is-a relation. The idea is based on the observation that WordNet senses are too fine-grained, therefore the neighbors share a lot of semantic meanings. To find similar senses, we use the relations: hypernym, hyponym, similar attributes, similar verb group, pertinym, holonym, and meronyms.[4] The algorithm assumes that the words in the input are POS tagged. It is worth noting the differences between our algorithm and the PEA05 algorithm, though we take our cue from it. In PEA05, the authors retrieve all the relevant neighbors to form a large bag of words for both the target sense and the surrounding sense and they specifically focus on the Lesk similarity measure. In our current work, we employ the neighbors in a disambiguation strategy using different similarity measures one pair at a time.

This algorithm takes as input a target sense and a sense pertaining to a word in the surrounding context, and returns a sense similarity score. It is worth noting that we do not apply the WN relations expansion to the target sense. It is only applied to the contextual word. We experimented with expanding both the contextual sense and the target sense and we found that the unreliability of some of the relations is detrimental to the algorithm's performance. Hence we decided empirically to expand only the contextual word.

We employ the same normalization values used in SM07 for the different similarity measures. Namely for the Lesk and Expand-Lesk we use the same cut off value of 240, accordingly, if the Lesk or Expand-Lesk similarity value returns $0 <= 240$ it is con-

[3] These values were decided on based on calibrations on the SENSEVAL 3 data set.

[4] We have run experiments varying the number of relations to employ and they all yielded relatively similar results. Hence in this paper, we report results using all the relations listed above.

verted to a real number in the interval [0,1], any similarity over 240 is by default mapped to a 1. For JCN, similar to SM07, the values are from 0.04 to 0.2, we mapped them to the interval [0,1]. It is worth noting that we did not run any calibration studies beyond the what was reported in SM07.

SemCor Expansion of WordNet A basic part of our approach relies on using the Lesk algorithm. Accordingly, the availability of glosses associated with the WordNet entries is extremely beneficial. Therefore, we expand the number of glosses available in WordNet by using the SemCor data set, thereby adding more examples to compare. The SemCor corpus is a corpus that is manually sense tagged (Miller, 1990). In this expansion, depending on the version of WordNet, we use the sense-index file in the WordNet Database to convert the SemCor data to the appropriate version sense annotations. We augment the sense entries for the different POS WordNet databases with example usages from SemCor. The augmentation is done as a look up table external to WordNet proper since we did not want to dabble with the WordNet offsets. We set a cap of 30 additional examples per synset. Many of the synsets had no additional examples. A total of 26875 synsets in WordNet 1.7.1 and a total of 25940 synsets are augmented with SemCor examples.[5]

5 Experiments and Results

5.1 Data

We experiment with all the standard data sets, namely, Senseval 2 (SV2) (M. Palmer and Dang, 2001), Senseval 3 (SV3) (Snyder and Palmer, 2004), and SEMEVAL (SM) (Pradhan et al., 2007) English All Words data sets. We used the true POS tag sets in the test data as rendered in the Penn Tree Bank. We exclude the data points that have a tag of ”U” in the gold standard since our system does not allow for an unknown option (i.e. it has to produce a sense tag). We present our results on 3 versions of WordNet (WN), 1.7.1 for ease of comparison with previous systems, 2.1 for SEMEVAL data, and 3.0 in order to see whether the trends in performance hold across WN versions.

5.2 Evaluation Metrics

We use the `scorer2` software to report fine-grained (P)recision and (R)ecall and (F)-measure on the different data sets.

5.3 Baselines

We consider here the two different baselines. 1. A random baseline (RAND) is the most appropriate baseline for an unsupervised approach. We consider the first sense baseline to be a supervised baseline since it depends crucially on SemCor in ranking the senses within WordNet.[6] It is worth pointing out that our algorithm is still an unsupervised algorithm even though we use SemCor to augment WordNet since we do not use any annotated data in our algorithm proper. 2. The SM07 baseline which we consider our true baseline.

5.4 Experimental Conditions

We explore 4 different experimental conditions: JCN-V which uses JCN instead of LCH for verb-verb similarity comparison, we consider this our base condition; +ExpandL is adding the Lesk Expansion to the base condition; +SemCor adds the SemCor expansion to the base condition; and finally +ExpandL_SemCor, adds the latter both conditions simultaneously.

5.5 Results

Table 1 illustrates the obtained results on the three data sets reporting only overall F-measure. The coverage for SV2 is 98.36% losing some of the verb and adverb target words. The coverage for SV3 is 99.7% and that of SM is 100%. These results are on the entire data set as described in Table **??**. Moreover, Table 2 presents the detailed results for the Senseval 2 data set using WN 1.7.1 since it is the most studied data set and for ease of comparison with previous studies. We break the results down by POS tag (N)oun, (V)erb, (A)djective, and Adve(R)b.

[5] It is worth noting that some example sentences are repeated across different synsets and POS since the SemCor data is annotated as an All-Words tagged data set.

[6] From an application standpoint, we do not find the first sense baseline to be of interest since it introduces a strong level of uniformity – removing semantic variability – that is not desirable. Even if the frist sense achieves higher results in these data sets, it is an artifact of the size of the data and the very limited number of documents under investigation.

Condition	SV2-WN171	SV2-WN30	SV3-WN171	SV3-WN30	SM-WN2.1	SM-WN30
RAND	39.9	41.8	32.9	33.4		25.4
SM07	59.7	59.8	54	53.8	40.4	40.8
JCN-V	60.2	60.2	55.9	55.5	44.1	45.5
+ExpandL	60.9	60.6	55.7	55.5	43.7	45.1
+SemCor	62.04	62.2	**59.7**	**60.3**	**46.8**	**46.8**
+ExpandL_SemCor	**62.7**	**62.9**	59.5	59.6	45.9	45.7

Table 1: F-measure % for all experimental conditions on all data sets

Condition	N	V	A	R
RAND	43.7	21	41.2	57.4
SM07	68.7	33.01	65.2	63.1
JCN-V	68.7	35.46	65.2	63.1
+ExpandL	**70**	35.86	65.6	62.8
+SemCor	68.3	37.86	**68.6**	68.75
+ExpandL_SemCor	69.5	**38.66**	68.2	**69.15**

Table 2: F-measure results per POS tag per condition for SV2 using WN 1.7.1.

6 Discussion

Our overall results on all the data sets clearly outperform the baseline as well as state of the art performance using an unsupervised system (SM07) in overall accuracy across all the data sets. Our implementation of SM07 is slightly higher than those reported in (Sinha and Mihalcea, 2007), 57.12% is probably due to the fact that we do not consider the items tagged as ”U” and also we resolve some of the POS tag mismatches between the gold set and the test data. We note that for the SV2 data set our coverage is not 100% due to some POS tag mismatches that could not have been resolved automatically. These POS tag problems have to do mainly with multiword expressions and the like.

In observing the performance of the overall system, we note that using JCN for verbs clearly outperforms using the LCH similarity measure across the board on all data sets as illustrated in Table 1. Using SemCor to augment WordNet examples seems to have the biggest impact on SV3 and SM compared to ExpandL. This may be attributed to the fact that the percentage of polysemous words in the latter two sets is much higher than it is for SV2. Combining SemCor with ExpandL yields the best results for the SV2 data sets. There seems to be no huge notable difference between the three versions of WN, though WN3.0 seems to yield slightly higher results maybe due to higher consistency in the overall structure when comparing WN1.7.1, WN2.1, and WN3.0. We do recognize that we can’t directly compare the various WordNets except to draw conclusions on structural differences remotely. It is also worth noting that less words in WN3.0 used SemCor expansions.

Observing the results yielded per POS in Table 2, ExpandL seems to have the biggest impact on the Nouns only. This is understandable since the nouns hierachy has the most dense relations and the most consistent ones. SemCor augmentation of WN seemed to benefit all POS significantly except for nouns. In fact the performance on the nouns deteriorated from the base condition JCN-V from 68.7 to 68.3%. This maybe due to inconsistencies in the annotations of nouns in SemCor or the very fine granularity of the nouns in WN. We know that 72% of the nouns, 74% of the verbs, 68.9% of the adjectives, and 81.9% of the adverbs directly exploited the use of SemCor augmented examples. Combining SemCor and ExpandL seems to have a positive impact on the verbs and adverbs, but not on the nouns and adjectives. These trends are not held consistently across data sets. For example, we see that SemCor augmentation helps both all POS tag sets over using ExpandL alone or when combined with SemCor. In order to analyze this further, we explore the performance on the polysemous POS only in all the data sets. We note that the same trend persists, SemCor augmentation has a negative impact on the SV2 data set in both WN 1.7.1. and WN 3.0. yet it benefits all POS in the other data sets, namely SV3 WN1.7.1 and SV3 WN3.0, SM WN2.1 and SM WN3.0.

We did some basic data analysis on the items we are incapable of capturing. Several of them are cases of metonymy in examples such as ”the English are known...”, the sense of *English* here is clearly in reference to the people of England, however, our WSD system preferred the language sense of the word. If it had access to syntactic/semantic role we would assume it could capture that this sense of the word entails volition for example. Other types of errors resulted from the lack of a method to help identify multiwords.

7 Conclusions and Future Directions

In this paper, we presented improvements on state of the art monolingual all words WSD using a well established graph based algorithm coupled with enhancements on basic similarity measures. We also explored the impact of augmenting WordNet with more gloss examples from a hand annotated resource as a means of improving WSD performance. We present the best results to date for an unsupervised approach on standard data sets: Senseval 2 (62.7%) using WN1.7.1, and Senseval 3 (59.7%) using WN1.7.1. In the future, we would like to explore the incorporation of multiword chunks, document level lexical chains, and syntactic features in the modeling of the Lesk overlap measure. We would like to further explore why ExpandL conditions did not yield the expected high performance across the different POS tags. Moreover, we are still curious as to why SemCor expansion did not help the nouns performance in SV2 conditions specifically.

References

Marine Carpuat and Dekai Wu. 2007. Improving statistical machine translation using word sense disambiguation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 61–72, Prague, Czech Republic, June. Association for Computational Linguistics.

Yee Seng Chan, Hwee Tou Ng, and David Chiang. 2007. Word sense disambiguation improves statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 33–40, Prague, Czech Republic, June. Association for Computational Linguistics.

Christiane Fellbaum. 1998. ”wordnet: An electronic lexical database”. MIT Press.

J. Jiang and D. Conrath. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of the International Conference on Research in Computational Linguistics*, Taiwan.

C. Leacock and M. Chodorow. 1998. Combining local context and wordnet sense similarity for word sense identification. In *WordNet, An Electronic Lexical Database*. The MIT Press.

M. Lesk. 1986. Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. In *In Proceedings of the SIGDOC Conference*, Toronto, June.

S. Cotton L. Delfs M. Palmer, C. Fellbaum and H. Dang. 2001. English tasks: all-words and verb lexical sample. In *In Proceedings of ACL/SIGLEX Senseval-2*, Toulouse, France, June.

Rada Mihalcea. 2005. Unsupervised large-vocabulary word sense disambiguation with graph-based algorithms for sequence data labeling. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 411–418, Vancouver, British Columbia, Canada, October. Association for Computational Linguistics.

George A. Miller. 1990. Wordnet: a lexical database for english. In *Communications of the ACM*, pages 39–41.

Roberto Navigli. 2009. Word sense disambiguation: a survey. In *ACM Computing Surveys*, pages 1–69. ACM Press.

Banerjee Pedersen and Patwardhan. 2005. Maximizing semantic relatedness to perform word sense disambiguation. In *University of Minnesota Supercomputing Institute Research Report UMSI 2005/25*, Minnesotta, March.

Sameer Pradhan, Edward Loper, Dmitriy Dligach, and Martha Palmer. 2007. Semeval-2007 task-17: English lexical sample, srl and all words. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 87–92, Prague, Czech Republic, June. Association for Computational Linguistics.

Ravi Sinha and Rada Mihalcea. 2007. Unsupervised graph-based word sense disambiguation using measures of word semantic similarity. In *Proceedings of the IEEE International Conference on Semantic Computing (ICSC 2007)*, Irvine, CA.

Benjamin Snyder and Martha Palmer. 2004. The english all-words task. In Rada Mihalcea and Phil Edmonds, editors, *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pages 41–43, Barcelona, Spain, July. Association for Computational Linguistics.