ISSN 2395-8618

One Sense per Discourse Heuristic for Improving Precision of WSD Methods based on Lexical Intersections with the Context

Grigori Sidorov and Francisco Viveros-Jiménez

***Abstract*—Word sense disambiguation is the task of choosing a sense for a target word in a given text using some words from the text and, in some cases, hand-tagged samples or dictionary definitions. The sense list is taken usually from an explanatory dictionary for a given language. Note that since the word is part of the text, we rely on the context words for making the decision. The methods that use information from words in the (near) context are very simple, because they consider lexical intersections of the word with the context words and/or their definitions or samples of usage. These methods reach precision of up to 70%. There are also methods that have better performance, but they are much more sophisticated: they use expensive resources – usually hand crafted – and rely on complex algorithms. In this paper, we show how to increase precision for certain word classes of these simple methods to the level comparable with that of the most sophisticated ones. Namely, we observed that these methods usually disambiguate correctly those words that conform to the One Sense per Discourse heuristic (OSD words). We used Semcor and Wikipedia to find the OSD words and left non-OSD words without disambiguation, thus improving precision at the expense of recall. Our motivation for this situation – more precision, less recall – is: (1) if we need high quality disambiguation and use human evaluators, then we can reduce the cost by asking them to disambiguate only words that are really difficult for the algorithms; (2) in an automatic system, we can apply this method for disambiguation of the corresponding words, and use other more sophisticated method for disambiguation of other words, i.e., use different methods for disambiguation (meta-disambiguation). We experimented with the complete and simplified Lesk algorithms, the graph based algorithm, and the first sense heuristic. The precision of all algorithms increases and some algorithms reach the level of the inter annotator agreement.**

***Index Terms*—Word sense disambiguation, one sense per discourse heuristic, context, lexical intersections.**

I. Introduction

WORDS have different meanings depending on the context. For example, in the sentence “John is drawing a **tree**”, the last word can mean a plant or a graph. Word sense disambiguation (WSD) is the task of identifying the sense (meaning) of a target word in a context [1]. Word senses are taken from a specific explanatory dictionary.

Generally speaking, WSD is a complex problem, which may require for its solution application of various methods of artificial intelligence. Currently, there are numerous solutions for tackling WSD. Simple methods that are based on the knowledge about the word itself and the words in its context have relatively low performance (the best methods obtain precision of about 60%). More complex supervised methods can reach precision above 70% [2], [3]. Still, these supervised methods need manually tagged training data, which is expensive and in real life is not always affordable.

WSD is useful for many NLP applications that deal with the meaning of texts, such as machine translation [4], [5], [6], wikification [7], information retrieval [8], etc. So there is a need in WSD systems with high precision when designing systems for these tasks. Note that a need in a reliable WSD system persists even if such system disambiguates only some target words (i.e., not all of them).

This paper describes the method that allows increasing precision of WSD systems at the expense of recall/coverage. The main idea is to disambiguate just those words that comply with the one sense per discourse (OSD) heuristic. Further this idea is analyzed in detail.

Previously, it was reported [9] that using features for selective disambiguation leads to a performance boost of about 5%. In that work the authors used word features, such as word grain, amount of positive and negative training examples and dominant sense ratio. They went even further and ensemble a back-off chain of three methods in a metaheuristic that selects the best method (of the three) using these word features. We propose to rely only on the One Sense per Discourse heuristic, but our precision boosts are greater than the ones reported by [9].

Our motivation for this situation – more precision, less recall– is: (1) if we need high quality disambiguation and use human evaluators, then we can reduce the cost by asking them to disambiguate only words that are really difficult for the algorithms; (2) in an automatic system, we can apply this method for disambiguation of the corresponding words, and use other more sophisticated method for disambiguation of

Manuscript received on March 15, 2016, accepted for publication on September 30, 2017, published on June 30, 2018.

Grigori Sidorov and Francisco Viveros-Jiménez are with the Centro de Investigación en Computación, Instituto Politécnico Nacional, Mexico City, Mexico (web: www.cic.ipn.mx/ sidorov, e-mail: pacovj@hotmail.com).

https://doi.org/10.17562/PB-57-4

ISSN 2395-8618

other words, i.e., use different methods for disambiguation (meta-disambiguation).

In the following sections, we describe the corresponding experiments and present a discussion about the behavior of the proposed method.

II. Experimental Setup

As we already mentioned, our hypothesis is that disambiguating only OSD words increases precision, but it obviously disambiguates fewer words, so recall and coverage become lower. Experiments were conducted for confirming this hypothesis. We show that this cost is acceptable, i.e., the WSD systems benefit from the proposed method. We also show that this phenomenon does not depend on the disambiguating algorithm, the test set data, and the word sense inventory.

We tested four algorithms over four test sets using different explanatory dictionaries (sense inventories). The explanatory dictionaries were: the WordNet 3.0 [10], Wikipedia [11] and Spanish Wikipedia. The test sets were: Senseval 2 [2], Senseval 3 [3], and hand-picked English/Spanish Wikipedia articles. The used WSD algorithms were: the Simplified Lesk algorithm [12], the Graph Indegree with Lesk measure [13], the traditional Lesk algorithm [14] and the first sense heuristic [12]. We also present additional experiments with Conceptual Density [15], Naive Bayes [16] and GETALP [17], [18] for some test sets.

We used precision (P), recall (R), coverage (C) and F-measure (F1) for measuring the performance of the algorithms as specified in [1]. They were calculated with the following equations:

$$\begin{aligned} \text{P} &= \frac{\text{correct answers}}{\text{answers}}, \\ \text{R} &= \frac{\text{correct answers}}{\text{words}}, \\ \text{C} &= \frac{\text{answers}}{\text{words}}, \\ \text{F1} &= \frac{2 \times P \times R}{P + R}, \end{aligned}$$

where *answer* is the target word, for which the algorithm has selected a sense, and *correct answer* is the answer that coincides with the one provided by human annotators as the gold standard.

We compare the performance of the algorithms in three different situations:

1) Disambiguating each sentence independently.
2) Forcing the algorithms to use the one sense per discourse (OSD) assumption [19], [20]. In this case, the WSD algorithms disambiguate all instances of the target word independently using the corresponding sentences as the context.
3) Disambiguating only words that usually comply with the one sense per discourse heuristic (OSD).

A. Wikipedia Test Set

Besides using Senseval 2 and Senseval 3 data, we also chose 12 Wikipedia articles in Spanish and English languages as our empirical test set. Note that articles in Spanish Wikipedia are generally shorter and have less polysemy than their counterparts in English, as can be seen in Table I. These differences make disambiguating Spanish articles easier.

B. About our Implementation

We used Java as our main programming language and a computer with Intel Core I3 with 4GB in RAM for testing. Our Java implementation is available for using under a non-commercial license at *http://sourceforge.net/gannu*. It contains command line and graphical tools for performing the following tasks:

- Setting up the experiments.
- Running the experiments (test results are stored into XLS files).
- Searching for sense definitions.
- Creating gold standard files from raw text and Wikipedia articles.
- Loading samples into a dictionary.

The package also contains complete API documentation and tutorials.

C. Implementation Details Related with the WordNet

We used both glosses and samples as the base definitions. We discarded stop-words using the predefined list. The Stanford POS tagger [21] and the lemmatizer based on WordNet [10] were used for generation of the final definitions of words (word senses). Our test results differ from the reported results – less than ±3% in F1-measure – because we used the different – the latest – version of the WordNet. Note that our results can be easily reproduced, because the source code and the data are available.

D. Implementation Details Related with Wikipedia

We used the first paragraphs of Wikipedia definitions, which appear before the table of contents or a section mark of articles, as definitions of word senses. We used the manually inserted hyperlinks in the articles and the disambiguation pages as our gold standard. For example, if we want to disambiguate the word *Wolf*, a WSD system have to select between the 40 senses listed in the *wiki/Wolf_(disambiguation)* page. If this word is tagged with the hyperlink *wiki/Gray_wolf*, then we know that the correct sense is the one corresponding to this link.

III. Calculation of the One Sense per Discourse Condition

Some words very often comply with the one sense per document (OSD) heuristic, which tells us that these words

TABLE I
SELECTED WIKIPEDIA ARTICLES AND SOME OF THEIR FEATURES.

	English			**Spanish**		
	Running	Target	Polysemy	Running	Target	Polysemy
Book	7152	406	7.9	3471	228	4.7
Calculator	6965	259	11.0	5837	202	1.8
Chemistry	8619	748	7.4	2849	223	3.7
Computer	9386	694	11.7	2885	195	3.0
Dog	15113	541	9.9	10161	366	4.7
Gray Wolf	16667	933	15.7	9438	289	6.1
Iron Man	11288	499	7.0	7371	225	3.3
Penicillin	4603	240	5.2	12087	692	2.9
Printing Press	6593	257	8.2	2492	86	4.9
Science	11123	753	10.4	17086	620	3.4
Spider-Man	9626	481	9.2	8519	239	4.5
Tiger	16744	639	13.1	4667	249	6.7
Average	**10323**	**537**	**10.4**	**7239**	**301**	**4.0**

usually have single meaning inside a text [19]. The OSD heuristic was successfully used for disambiguating some selected nouns in [20]. However, it was reported that not all words comply with this heuristic [22], so it is not a good idea to apply the OSD heuristic for all words (as we confirm later in this research).

We used two procedures for calculating if a word complies with the OSD heuristic or not. For WordNet based tests, we used the SemCor corpus [23]. A word complies with the OSD heuristic if it appears in this corpus with exactly one sense per document or it does not exist in the corpus.

For Wikipedia based tests, we used Wikipedia search counts. These search counts are stored in a matrix containing the search hits of all possible pair of senses, i.e. , our algorithm searches for the frequency of co-occurences of senses. Thus, each matrix element stores the frequency of a pair of senses. Diagonal elements contain the search counts of single senses. All counts are decreased by an empirical value of $2\times Polysemy$ hits due to the existence of disambiguation, category and list pages which contain sense pairs of the same word. A word complies with the OSD heuristic when all of the non-diagonal element of the matrix are less or equal to zero.

IV. Performance Analysis

Test results presented in Tables II, III, and V confirm that disambiguating just the words that comply with the OSD heuristic increases precision at the cost of recall/coverage. The precision boost was from 3% to 25%, being the average of 16%. The coverage loss was from 11% to 57%, being the average of 34%. Also, we observed that the first sense heuristic together with the OSD heuristic had the best approach in the tests: it obtained precision in the range from 79% to 99%. Note that forcing the OSD heuristic assumption does not lead to a consistent increase in precision (although, it often leads to a coverage boost). For further reference we added a table containing the best results observed in Senseval 2 and Senseval 3, see Table IV. Note that the first sense heuristic together with the OSD heuristic overcame the precision of the best systems in these competitions.

Figure 1 provides a graphical representation of the performance changes. This figure shows the precision obtained for different values of coverage. Coverage can be changed by using different window sizes in the range of [1,1024] (i.e., window size values are directly related to coverage). This figure confirms that all selected algorithms solve better the OSD words. Hence, our OSD filter allows other algorithms to outperform first sense heuristic. Moreover, the Graph InDegree algorithm was able to get the value of the inter annotator agreement for the OSD words.

TABLE II
TEST RESULTS CORRESPONDING TO GETALP SYSTEM OBSERVED IN SEMEVAL 2013 AND SENSEVAL 3 COMPETITIONS.

Semeval 2013 [18]	P	R	C	F1
GETALP with OSD	65.7	37.9	57.6	48.1
GETALP	51.6	51.6	100	51.6

V. Words that do not Comply with the OSD Heuristic

We analyzed the words that do not comply with the OSD heuristic. There are words of all grammar classes, as seen in Table VII. The amount of such words is in the range from 14% to 58%. Most of the words that comply with the OSD heuristic are domain words like *scientist, cell, cancer, strategy, treatment*, etc.

Words that do not comply with the OSD heuristic have at least one of these traits:

- their sense definitions are similar between themselves,
- their sense definitions have very few (usually, less than three) open-class words, and
- their meaning is related to their current syntactic functions rather than to a possible document domain.

ISSN 2395-8618

TABLE III
TEST RESULTS CORRESPONDING TO CONCEPTUAL DENSITY AND NAIVE BAYES ALGORITHMS OBSERVED IN SENSEVAL 2 AND SENSEVAL 3 COMPETITIONS.

	Senseval 2				Senseval 3			
	P	R	C	F1	P	R	C	F1
Conceptual Density with OSD	**57.1**	5.8	10.0	10.5	**64.7**	13.4	20.7	22.2
Conceptual Density	49.2	9.7	19.8	16.2	51.7	34.8	67.2	41.6
Naive Bayes with OSD	**73.7**	36.0	48.9	48.3	**74.5**	30.6	41.1	43.4
Naive Bayes	58.4	57.0	97.6	57.7	54.9	54.2	98.9	54.6

TABLE IV
SYSTEMS HAVING THE HIGHEST PRECISION IN SENSEVAL 2 AND SENSEVAL 3 COMPETITIONS.

Senseval 2	P	R	C	F1
First Sense with OSD	**78.8**	40.0	50.9	53.1
IRST [24]	74.8	35.7	47.7	48.3
SMUaw [25]	69.0	69.0	100	69.0
CNTS-Antwerp [26]	63.6	63.6	100	69.0
Senseval 3				
First Sense with OSD	**79.3**	33.1	42.3	46.5
IRST-DDD-09-U [27]	72.9	44.1	60.5	54.9
IRST-DDD-LSA-U [27]	66.1	49.6	75.0	56.6
Gambl-AW-S [28]	65.1	65.1	100	65.1

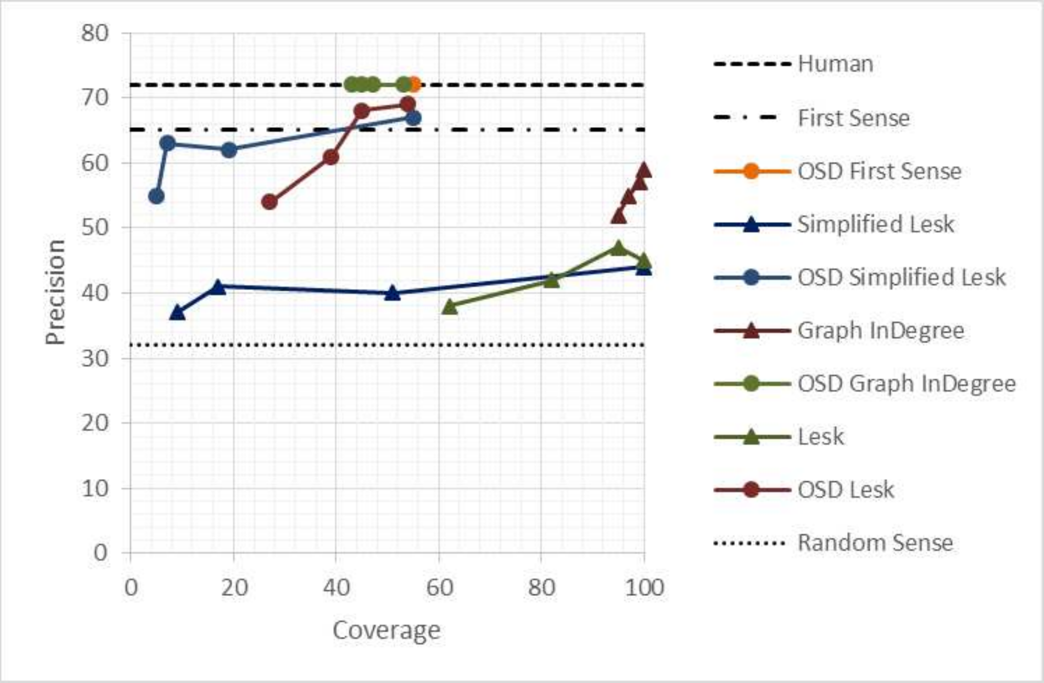

Fig. 1. Precision/coverage graph for some knowledge-based algorithms observed on Senseval 2 test set. Algorithms using our OSD filter (circles) overcame the first sense heuristic precision. Also, some algorithms overcame the human annotator agreement.

Table VI contains some sample definitions that are too similar to distinguish between them or too short for WSD systems.

The most discarded words are verbs. Common verbs (like *be, have* and *do*) have more than ten definitions in WordNet and are used widely across all domains. Often the main part of the meaning of verbs is heavily related to its complements. Take for example the following text: "*I started drinking some soda. Later, I decided to drink a cold beer.*" and the following definitions [$drink_V^1$:take in liquids] and [$drink_V^2$:consume alcohol] extracted from the WordNet. In this example, both definitions are clear for people but they are rather short for WSD algorithms. Also, the verb *drink* does not comply with the OSD heuristic. Furthermore, we can easily select the sense

TABLE V
PERFORMANCE COMPARISON OF SOME BAG OF WORDS ALGORITHMS. ALL METHODS EXHIBIT A PRECISION BOOST AND A COVERAGE LOSS WHEN SOLVING JUST WORDS THAT COMPLY WITH THE OSD HEURISTIC. **ST** MEANS DISAMBIGUATION USING SENTENCE WORDS, **FG** MEANS FORCING OSD FOR ALL WORDS, AND **OSD** MEANS SOLVING JUST WORDS THAT COMPLY WITH THE OSD HEURISTIC.

Senseval 2												
	First sense			Simplified Lesk			Graph InDegree			Lesk		
	St	Fg	OSD	St	Fg	OSD	St	Fg	OSD	St	Fg	OSD
P	67.1	67.1	78.8	39.5	45.5	61.0	59.7	57.5	78.1	48.1	49.4	67.6
R	67.1	67.1	40.0	19.6	31.0	12.1	59.6	57.4	39.9	46.0	49.4	31.9
C	100	100	50.9	49.7	68.1	19.8	99.8	99.9	51.1	95.6	99.9	47.2
F1	67.1	67.1	53.1	26.2	36.8	20.2	59.6	57.4	52.9	47.0	49.4	43.3
Senseval 3												
P	66.1	66.1	79.3	30.3	30.9	52.7	50.5	51.1	70.1	38.4	41.0	64.3
R	66.1	66.1	33.1	19.5	23.3	10.4	50.1	50.9	29.5	36.7	40.8	24.3
C	100	100	42.3	64.4	75.5	19.8	99.2	99.7	42.1	94.2	99.4	37.8
F1	66.1	66.1	46.5	23.7	26.6	17.4	50.3	51.0	41.5	37.8	40.9	35.2
English Wikipedia												
	First sense			Simplified Lesk			Graph InDegree			Lesk		
	St	Fg	OSD	St	Fg	OSD	St	Fg	OSD	St	Fg	OSD
P	89.5	89.2	95.8	71.5	73.5	92.3	72.5	72.9	92.0	70.3	68.8	92.9
R	89.5	89.2	68.5	57.6	62.6	50.1	66.7	69.2	58.8	49.9	65.8	46.3
C	100	100	71.5	80.6	85.1	54.2	92.1	94.9	63.9	71.0	95.5	49.9
F1	89.5	89.2	79.9	63.8	67.6	64.9	69.5	71.0	71.8	58.4	67.2	61.8
Spanish Wikipedia												
P	96.3	96.3	99.6	87.2	87.4	98.7	87.0	87.0	98.5	85.3	84.8	98.1
R	96.3	96.3	85.5	76.1	79.2	72.5	80.3	82.2	76.9	59.4	66.6	57.2
C	100	100	85.8	87.2	90.6	73.4	92.4	94.4	78.1	69.7	78.6	58.3
F1	96.3	96.3	92.0	81.3	83.1	83.6	83.5	84.5	86.4	70.0	74.6	72.3

TABLE VI
SOME DEFINITIONS THAT ARE TOO SIMILAR (TOP) OR SHORT (BOTTOM).

$Medical_J^1$	relating to the study or practice of medicine
$Medical_J^2$	requiring or amenable to treatment by medicine as opposed to surgery
$Bell_N^5$	the shape of a bell
$Recent_J^1$	new

TABLE VII
AVERAGE WORDS DISCARDED OF EACH CLASS.

	Noun	Verb	Adjective	Adverb
Senseval 2	39%	74%	43%	50%
Senseval 3	47%	81%	38%	0%
English Wiki	28%	–	–	–
Spanish Wiki	14%	–	–	–

of the verb *drink* by looking at the direct object in both cases. It is typical lexical function. Hence, in our future research we will try to design a system for disambiguating these words by using syntactic information.

VI. CONCLUSIONS

This paper shows that WSD methods can attain high precision when solving just those words that comply with one sense per discourse heuristic at the cost of losing recall/coverage. The precision boost is high enough to overcome the first sense baseline: this achievement can be reached only by complex state-of-the-art WSD systems. Also, our experimental results show that words that do not comply with OSD have one of these traits: (1) their meaning depends on the sentence rather than the domain (like most of the verbs), and, (2) their sense definitions are not adequate for current systems (they are too short or too similar between them).

We recommend disambiguating just the OSD words for increasing precision of WSD algorithms for real life applications requiring high precision.

REFERENCES

[1] R. Navigli, “Word sense disambiguation: A survey,” *ACM Comput. Surv*, vol. 41, no. 2, pp. 10:1–10:69, 2009.

[2] M. Palmer, C. Fellbaum, S. Cotton, L. Delfs, and H. T. Dang, “English tasks: All-words and verb lexical sample,” in *Proc. of ACL/SIGLEX Senseval-2*, 2001.

[3] B. Snyder and M. Palmer, “The English all-words task,” in *Proc. of ACL/SIGLEX Senseval-3*, 2004.

[4] Y. S. Chan and H. T. Ng, “Word sense disambiguation improves statistical machine translation,” in *Proc. of ACL 2007*, 2007, pp. 33–40.

[5] M. Carpuat, Y. Shen, X. Yu, and D. Wu, “Toward Integrating Word Sense and Entity Disambiguation into Statistical Machine Translation,” in *Proc. of IWSLT*, 2006, pp. 37–44.

ISSN 2395-8618

[6] D. Pinto, C. Balderas, M. Tovar, and B. Beltran, “Evaluating n-gram models for a bilingual word sense disambiguation task,” *Computación y Sistemas*, vol. 15, no. 2, 2011.

[7] R. Mihalcea and A. Csomai, “Wikify!: Linking documents to encyclopedic knowledge,” in *Proc. of CIKM 2007*, 2007, pp. 233–242.

[8] C. Stokoe, M. P. Oakes, and J. Tait, “Word sense disambiguation in information retrieval revisited,” in *Proc. of the 26th annual international ACM SIGIR conference on Research and development in information retrieval*, 2003, pp. 159–166.

[9] H. M. Saarikoski, S. Legrand, and A. Gelbukh, “Defining classifier regions for WSD ensembles using word space features,” in *MICAI 2006: Advances in Artificial Intelligence*, 2006, pp. 885–867.

[10] G. A. Miller, “A lexical database for English,” *Communications of the ACM*, vol. 38, pp. 39–41, 1995.

[11] Wikipedia, “Wikipedia: The free encyclopedia,” 2004.

[12] A. Kilgarriff and J. Rosenzweig, “Framework and results for English SENSEVAL,” *Computers and the Humanities*, vol. 34, no. 1-2, pp. 15–48, 2000.

[13] R. Sinha and R. Mihalcea, “Unsupervised graph-based word sense disambiguation using measures of word semantic similarity,” in *Proc. of ICSC 2007*, 2007, pp. 363–369.

[14] M. Lesk, “Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone,” in *Proc. of SIGDOC*, 1986, pp. 24–26.

[15] D. Buscaldi, P. Rosso, and F. Masulli, “Finding predominant word senses in untagged text,” in *Workshop Senseval-3, Proc. of ACL*, 2004, pp. 77–82.

[16] D. Yuret, “Some experiments with a Naive Bayes WSD system,” in *Proc. of ACL/SIGLEX Senseval-3*, 2004.

[17] D. Schwab, J. Goulian, and A. Tchechmedjiev, “Worst-case complexity and empirical evaluation of artificial intelligence methods for unsupervised word sense disambiguation,” *International Journal of Web Engineering and Technology*, vol. 8, no. 2, pp. 124–153, 2013.

[18] R. Navigli, D. Jurgens, and D. Vannella, “Semeval-2013 task 12: Multilingual word sense disambiguation,” in *Proceedings of the 7th International Workshop on Semantic Evaluation*, 2013, pp. 222–231.

[19] W. A. Gale, K. W. Church, and D. Yarowski, “One sense per discourse,” in *Proc. of HLT*, 1991, pp. 233–237.

[20] D. Yarowsky, “Unsupervised word sense disambiguation rivaling supervised methods,” in *Proc. of ACL 2007*, 1995, pp. 189–196.

[21] K. Toutanova and C. D. Manning, “Enriching the knowledge sources used in a maximum entropy part-of-speech tagger,” in *Proc. of EMNLP*, 2000, pp. 63–70.

[22] D. Martínez and E. Agirre, “One Sense per Collocation and Genre/Topic Variations,” in *Proc. of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, 2000.

[23] G. A. Miller, M. Chodorow, S. Landes, C. Leacock, and R. G. Thomas, “Sing a semantic concordance for sense identification,” in *Proc. of ARPA Human Language Technology Workshop*, 1994, pp. 240–243.

[24] B. Magnini, C. Strapparava, G. Pezzulo, and A. Gliozzo, “Using domain information for word sense disambiguation,” in *Proc. of ACL/SIGLEX Senseval-2*, 2001.

[25] R. Mihalcea and D. Moldovan, “Pattern learning and active feature selection for word sense disambiguation,” in *Proc. of ACL/SIGLEX Senseval-2*, 2001.

[26] V. Hoste, A. Kool, and W. Daelemans, “Classifier optimization and combination in the english all words task,” in *Proc. of ACL/SIGLEX Senseval-2*, 2001.

[27] C. Strapparava, A. Gliozzo, and C. Giuliano, “Pattern abstraction and term similarity for word sense disambiguation: IRST at senseval-3,” in *Proc. of ACL/SIGLEX Senseval-3*, 2004.

[28] B. Decadt, V. Hoste, W. Daelemans, and A. Van den Bosch, “Genetic Algorithm Optimization of Memory-Based WSD,” in *Proc. of ACL/SIGLEX Senseval-3*, 2004.