On Wordsense Disambiguation Through Morphological Transformation and Semantic Distance and Domain link knowledge*

M. Oussalah
Centre for Ubiquitous Computing,
Faculty of Information technology,
University of Oulu,
Oulu, Finland.
Mourad.Oussalah@oulu.fi

M. Väisänen
Centre for Ubiquitous Computing,
Faculty of Information technology,
University of Oulu,
Oulu, Finland.

E. Gilman
Centre for Ubiquitous Computing,
Faculty of Information technology,
University of Oulu,
Oulu, Finland.

Abstract *- Despite the advances in information processing systems, word-sense disambiguation tasks are far to be satisfactory as testified by numerous limitations of current translation systems and text inference systems. This paper attempts to investigate new techniques in knowledge based word-sense disambiguation field. First, by exploring the WordNet lexical database and part-of-speech conversion through the established CatVar database that translates all non-noun words into their noun counterparts, and following the spirit of Lesk's disambiguation algorithm, a new disambiguation algorithm that maximizes the overall semantic similarity in the sense of Wu and Palmer measure between each sense of the target word and synsets of words of the context, is established. Second, motivated by the existence of WordNet domains for individual synsets, an overlapping based approach that quantifies the set intersection of synset domains, if not empty, or the hierarchy structure of the domains links through a simple path-length measure is put forward. Third, instead of exploring the whole set of words involved in the context, a selective approach that uses syntactic feature as outputted by Stanford Parser and a fixed length windowing is developed. The developed algorithms are evaluated according to two commonly employed dataset where a clear improvement to the baseline algorithm has been acknowledged.*

Keywords: Text mining, Word Sense Disambiguation, Semantic Similarity

I. INTRODUCTION

Word sense ambiguity is inherent and prevalent to human language where a single word can convey multiple meanings. For instance, "bank" may stand for a financial institution, objects (materials) grouped together in rows, high mass/mound of a particular substance, or a land near river / lake. The correct sense of an ambiguous word is typically selected based on the context where it occurs [1].

The appropriate handling of word sense disambiguation (WSD) task ultimately provides a major breakthrough in the design of effective information retrieval systems where identifying the correct sense of query terms, for instance, increases the performance of the query-documents matching process. Similarly, this also enhances the accuracy of machine translation systems, question-answering systems, among others. Especially, this would provide an edge towards what constitutes a fundamental contribution to the realization of the so-called semantic Web, "an extension of the current Web, in which information is given well-defined meaning, better enabling computers and people to work in cooperation" [2].

With the availability of large and reliable source of dictionary glosses and concepts (e.g., thesauri, Collins English Dictionary, Oxford Dictionary of English, WordNet, Omega ontology) as well as the cost-effective computational resources, the task of word sense disambiguation becomes attractive. On the other hand, motivated by corpus based linguistic research, especially, Senseval/Semeval evaluation competitions[1], several manually annotated corpora, have been put forward. This promoted the development of new algorithms for disambiguation task. These algorithms were evaluated on pre-defined corpora and benchmarked against manually annotated 'gold standards' of the same corpus.

Besides the supervised and unsupervised based techniques, one distinguishes two streams of techniques for word-sense disambiguation: corpus-based and knowledge based methods. The former relies on the availability of sense-

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

tagged text that will be used to train a multitude of classifiers in order to predict the appropriate sense in a given context. The second stream relies on identifying a shared vocabulary between the definitions of words in the same spirit of Lesk algorithm [3]. In this course, Banerjee and Pederson [4] have suggested an adapted Lesk algorithm that uses WordNet as a source of glosses.

This paper advocates a knowledge based approach where new variants of enhanced Lesk algorithm will be put forward, according to the intuitive interpretations ascribed to the intersections of the various synsets. Especially, two types of interpretations are considered. The former assumes a metric viewpoint calculated using the path-length measure of WordNet hierarchical synset structure. While the latter utilizes the WordNet domains links and extends both the set intersection and available domain hierarchical distance metric accordingly. The main contribution of the paper are twofold. First, in order to benefit from the dense hierarchical structure of noun-category in WordNet lexical database [5], a word morphology transformation is employed, which then serves as a basis for subsequent semantic similarity. This ultimately increases chance of hitting common sense. Second a syntactic features were employed in order to select the wording that will be used as part of context. This allows the automated system to account for words that maybe located far away from the standard windowing selection around the target word. The performances of the suggested algorithms are evaluated using Senseval-2 dataset where a systematic improvement over the baseline has been noticed. Section 2 of this paper provides background and related work. Our methodology is detailed in Section 3. Implementation issues are discussed in Section, while testing and exemplification are reported in Section 5.

II BACKROUND AND RELATED WORK

A pioneer work in word sense disambiguation is the Lesk algorithm [3]. More formally, let W_t, be a target word whose meaning is defined in the set $N_{W_t} = \{W_t^1, W_t^2, .. W_t^{n_t}\}$ where n_t corresponds to the number of distinct meanings of the target word, according to some dictionary, e.g., Oxford English dictionary, WordNet. Given a sentence /phrase S that contains W_t (t takes values in {0,..,m}), say

$$S = < W_0, W_1, W_2, W_3, \ldots, W_m >$$

W_t is assigned a sense W_t^k such that

$$\left| W_t^k \cap \bigcap_{i=0,m;\ i\neq t} W_i \right| = \max_j \left| W_t^j \cap \bigcap_{i=0,m;\ i\neq t} W_i \right| \quad (1)$$

Lesk's methodology has been extended in various directions. Banerjee and Pedersen [4] suggested to use the phrases that appear at each synset (sense) pertaining to individual word as a counterpart of glosses in expression employing all wording involved in describing the synsets of the target word. Agirre and Rigau [6] proposed to use a (WordNet-based) semantic similarity in order to identify the correct sense. The latter corresponds, for instance, to the sense that maximizes the overall semantic similarity:

$$W_t \mapsto W_t^k : \sum_{i=0,m\ \&\ i\neq t} Sim\left(W_t^k, W_i\right) = \max_j \sum_{i=0,m\ \&\ i\neq t} Sim\left(W_t^j, W_i\right) \quad (2)$$

where *Sim*(.,.) stands for one of semantic similarity measures [1, 5, 11] (Wu and Palmer's measure was adopted). This approach works only if the two words of the pair belong to the same part-of-speech. Mihalcea and Moldovan [7] extended this concept to pairs of different part-of-speech, especially for nouns-verbs connected via syntactic relations such as verb-object and noun-adverb. Agirre and Rigau [8] introduced the concept of "conceptual density" defined as the overlap between the semantic concept hierarchy C (root of the hierarchy) and words in the same context.

III METHOD

Our general approach involves four distinct phases:

Phase 1: A morphological transformation was employed in order to transform all non-noun category (identified through an initial part-of-speech tagging) into corresponding noun categories. For this purpose, we use the Categorial Variation Database (CatVar) [9-10]. The latter is a lexical resource of morphological derivations for English words sharing a common stem. The PoS conversion augmented with CatVar is accomplished by finding the database cluster containing the word to be converted replacing it with a target word. In case of multiple nouns that can be associated to the given word, the algorithm picks up the first noun that induces the smallest Edit distance with the original word, which favours transformations that preserve as much of the original wording as possible. On the other hand, it should also be noted that not all wording will be subject to a categorical conversion. For instance, stopword, non-standard characters, URLs, unknown abbreviations are filtered out as part of the pre-processing stage. Similarly, pronouns, named-entities for which no entry is found in WordNet are left unchanged.

Phase 2. The process of word sense disambiguation of the target word involves the following. First, translate non-noun senses into noun-sense using the aforementioned CatVar transformation, yielding for each W_i, $\{ N_i^1, N_i^2, \ldots, N_i^{n_i} \}$. Second, calculate, for each sense W^k (k=1,n) of target word W, its associated score:

$$Score(W^k) = \sum_i \max_j Sim(N^k, N_i^j) \quad (3)$$

where N^k stands for the noun-counterpart, if required, of the sense W^k of the target word, and $Sim(.,.)$ stands for Wu and Palmer semantic similarity measure [11]. The sense W^{k*} of the target word is then selected such that

$$W^{k*} = \arg\max_{W^k} \max_k Score(W^k) \quad (4)$$

Third, inspired by SSI (structural semantic interconnection) algorithm [12], the implementation of (3-4) can be rendered simple using an iterative process by first selecting words *S'* in *S* that are monosemous, say:

$S' = \{W_i: \text{senses}(W_i) = \{W_i^1, W_i^2, ..., W_i^{n_i}\}, n_i = 1\}$.

Then, the counterpart of (3) becomes

$$Score(W^k) = \sum_{j:W_j \in S'} Sim(N^k, N_j) \quad (5)$$

Expression (3-4) or their SSI implementations, if any, allows us to select the appropriate sense of the target word W that maximizes the overall semantic similarity in the sense of Wu and Palmer WordNet similarity measure with all words of the context sentence *S*.

Phase 3. Motivated by the existence of domain categorization in WordNet domains project[2], the key idea is to utilize such information in the disambiguation task. Strictly speaking, WordNet domain project contains more than 100,000 domain links, where individual synset of noun, verb or adjective is assigned one or more Subject Field Codes (e.g., $doctor_n^1$ is tagged with the Medicine domain), or domain labels similar to the field labels used in dictionaries (e.g., Medicine, Engineering or Architecture). The domain labels are based on the Dewey Decimal Classification system and are arranged into a topic hierarchy [13]. We hypothesize that synsets that share the largest number of domain links are likely to have matching senses. Otherwise, if no common domain exists, the synsets that share the closest common subsumer in domains hierarchy are assumed to have coherent senses. Using a more formal representation, for a given synset W_i^j (j=1 to n_i) of word W_i , let $D_i^j = \{d_{ij}^1, d_{ij}^2, .., d_{ij}^{l_{ij}}\}$, j=1 to n_i i=1 to *m*, be the set of domain links associated to synset W_i^j . Similarly, let $D_0^k = \{d_k^1, d_k^2, .., d_k^{p_k}\}$, k=1 to n, be the domain links associated to synset W_k of the target word *W*, then an alternative to semantic similarity based disambiguation is:

$$Score\_d(W^k) = \left| \bigcap_{i=0,m} \left(\bigcap_j D_i^j \right) \right| \quad (6)$$

Therefore, the sense k* is chosen so that

$$W^{k*} = \arg\max_{W^k} \max_k Score\_d(W^k) \quad (7)$$

In case where all cardinalities |. | in (6) vanishes because there is no common domain link, an alternative to cardinality would be to explore the hierarchical structure of the domain links and compute the path-length *dist (.,.)* of the underlying nodes, which draws some analogy with WordNet Wu and Palmer semantic similarity such that:

$$Score\_d'(W^k) = \min_{j,k} \sum_{i=0,m-1} dist(D_i^j, D_{i+1}^k) \quad (8)$$

Therefore, the associated sense is determined as:

$$W^{k*} = \arg\min_{W^k} \min_k Score\_d'(W^k) \quad (9)$$

Especially, (8-9) expressions are triggered only if expression (6) yields zero-value for all senses W^k.

Interestingly, domain links-based reasoning does not require the word-part of speech transformation because the domain links exist for various part-of-speech category, and provide a sound alternative approach to wordsense disambiguation. On the other hand, as far as our testing is concerned, one should notice that most of synsets are rather assigned one single domain link; therefore, the hierarchical distance based scoring function (8-9) is the most applied one in the subsequent reasoning.

Phase 4. So far, in both the semantic similarity based approach and domain links-based matching, the reasoning is performed such that the target word is mapped to all words of the context sentence. However, it is acknowledged that this is expensive and likely not efficient approach to identify the correct sense. For instance, authors in [4] advocated the use of a fixed length window around the target word. Although such approach is simple it may also lead to missing important contextual information conveyed by other words of the sentence situated away from the selected window. Therefore a trade-off between window length, if any, and information brought by other words of the sentence sounds interesting. Thereby, the idea put forward is to use the syntactic information conveyed by the sentence, e.g., verb-object and subject-verb relations, in order to select the words-list over which the semantic similarity and domain link based approaches will be applied. This would require to use a Parser (Stanford Parser[3] is employed in our study). From the output of the parser, we extracted various sets of features. First, we distinguished direct relations (words

[2] http://wndomains.fbk.eu/
[3] https://nlp.stanford.edu/software/lex-parser.shtml

linked directly in the parse tree) from indirect relations (words that are two or more dependencies apart in the syntax tree, e.g. heads of prepositional modifiers of a verb). For each relation we also store its inverse-relation. The relations are coded according to the Minipar codes[4]. Therefore, the idea pursued in this trade-off based strategy is to select all wording of the original context sentence S where the target word W is linked to, through any type of syntactic relation as highlighted by the Stanford Parser in addition to a window of fixed length three around the target word. The choice of a smaller length window is motivated by computational consideration and the hope that this will be compensated by potential new words involved in syntactic relation with target word. Therefore, given a sentence S which includes the target word W_0, say, $S=\{W_0, W_1, .., W_m\}$ and let us assume the following configuration (ordering) of the words:

$$< W_1,\ W_2\ W_0,\ W_3,\ W_4, W_5, .., W_m >$$

The new set of words S' that will be used in formulating (3), (5), (8) is given by

$$S' = \{W_1, W_2, W_3, W_4, W_5\} \cup \{W \in S : relation(W, W_0) \neq \varnothing\}$$

where *relation* (W, W_0) indicates that Word W is linked with W_0 through Parser syntactic relation.

IV. EVALUATION

A. Exemplification

We deliberately selected an example from SemCor dataset[5] where the context involves more than one sentence, recognized to be among the challenging cases for the disambiguation task.

Example. Disambiguation of 'bass' with context '"Though still a far cry from the lake's record 52-pound bass of a decade ago, "you could fillet these fish again, and that made people very, very happy." Mr. Paulson says'. Correct sense is freshwater_bass.n.01. The result of the various algorithms are displayed in Table 1. For comparison purpose, we also employed an open source that incorporates a set of alternative disambiguation techniques using various WordNet based semantic similarity measures, referred to PyWSD, which is freely available in Github repository[6].

Results shown in Table 1 indicate that all the variants of our developed word sense disambiguation algorithms managed to identify the correct sense in the context sentences, while many of the alternative algorithms, including baseline -Lesk algorithm- fails to do so. The results also indicate that sometimes the algorithms cannot provide any answer, which explains the empty slots in the tables.

Table 1. Disambiguation of Example 1

Algorithm	Synset
CatVar Sem. Sim	freshwater_bass.n.01
Domains links overlapping	
CatVar Sem Sim. & Syntactic	freshwater_bass.n.01
Domain links overlap & Syntactic	
Lesk (comparison)	sea_bass.n.01
Simple Lesk (PyWSD)	sea_bass.n.01
Adapted Lesk (PyWSD)	sea_bass.n.01
Cosine Lesk (PyWSD)	bass.n.06
Jcn Similarity (PyWSD)	sea_bass.n.01
Lch similarity (PyWSD)	sea_bass.n.01
Lin similarity (PyWSD)	
Path similarity (PyWSD)	bass.n.06
Res similarity (PyWSD)	
Wup similarity (PyWSD)	sea_bass.n.06

B Testing with SensEval dataset

We used the test data from English lexical sample task used in Senseval-2 [14] comparative evaluation of word sense disambiguation systems. It contains a total of 4,328 test instances divided among 29 nouns, 29 verbs and 15 adjectives. Each test instance contains a sentence with a single target word to be disambiguated, and one or two surrounding sentences that provide additional context. There is also a gold standard tagging that was created by human annotators, and we use this only to evaluate the results of our algorithm. For the assessment purpose we use the standard Precision and Recall metrics. We distinguish results related to nouns and verbs categories. The results are reported in Table 2.

D Discussion

From the preceding, one shall mention the following.

- Result on SensEval evaluation dataset demonstrates the feasibility and high performance of our developed wordsense disambiguation algorithms. The performance achieved by CatVar semantic similarity based approach as well as Catvar –semantic similarity with syntactic features outperform the baseline, which is here Lesk'a algorithm. It is easy to see that the performance of CatVar Semantic similarity based approach outperforms that of Lesk's algorithm by 9.6% in precision and 9,8% in Recall for noun category, and more than 170% improvement for both

[4] http://wiki.opencog.org/w/Dependency_relations
[5] http://web.eecs.umich.edu/~mihalcea/downloads.html#semcor
[6] https://github.com/alvations/pywsd

precision and recall for verb-category in case of SensEval dataset. While using the best performing algorithm (CatVar Semantic similarity with syntactic features) yields an improvement of (16%, 15%) for noun-category and (147%, 170%) for verb-category in case of SemEval dataset.

Table 2. Overall disambiguation results on SensEval 2 dataset

Algorithm	Noun(%)		Verbs (%)		Runtime (s)
	P	R	P	R	
CatVar Sem. Sim	69	66	55	48	0.061
Domains links overlapping	44	41	29	26	0.052
CatVar Sem & Syntactic	71	69	52	57	0.056
Domain links over & Synt	43	40	27	49	0.051
Lesk (comparison)	61	60	21	18	0.0049
Simple Lesk (PyWSD)	32	28	29	28	0.0143
Adapted Lesk (PyWSD)	34	29	23	27	0.0182
Cosine Lesk (PyWSD)	28	26	18	19	0.0191
Jcn Similarity (PyWSD)	33	23	19	21	6.5014
Lch similarity (PyWSD)	29	26	22	19	0.0631
Lin similarity (PyWSD)	21	24	19	18	5.342
Path similarity (PyWSD)	28	25	17	18	0.0243
Res similarity (PyWSD)	36	32	36	28	6.325
Wup similarity (PyWSD)	41	39	0	0	0.0153

Among the four algorithms introduced in this paper, the Catvar-semantic similarity together with syntactic feature shows a marginal improvement over Catvar semantic similarity algorithm. This can be explained by the fact that the use of Parser allows the system to pick up relevant wording outside the sentence boundary of fixed length.

The performance of domain links overlapping based approach does not seem to be particularly interesting, especially with respect to the baseline. We believe that although the intuition behind the domain links based approach seems intuitive and appealing, the domain hierarchy still require further improvement and refinement, especially with respect to established WordNet synsets. The disambiguation result for verb-category seems to be relatively low as compared to noun-category as both precision and recall evaluations barely exceed 50%, a phenomenon which has also been reported elsewhere in the literature, which, in our view, is partly due to the effect of part-of-speech conversion.

Although, we acknowledge the existence of algorithms that achieved a disambiguation score of over 70% on the same dataset. Nevertheless, since all reported algorithms achieving such accuracy, as far as we know, are based on supervisory-like methods, we deliberately discard such classes of algorithms and only focused on knowledge based class of algorithms.

VI. CONCLUSION

In this paper knowledge based word-sense disambiguation task has been revised and four types of new algorithms have been put forward. This uses CatVar part-of-speech conversion while following the spirit of Lesk disambiguation algorithm, a new disambiguation scheme that maximizes the overall semantic similarity in the sense of Wu and Palmer measure between each sense of the target word and synsets of words of the context, is established. Other algorithms explore the WordNet domain links, restricting the context using syntactic features and a fixed length windowing. The developed algorithms have been tested using SemEval dataset. Especially, it is shown that the syntactic feature Catvar semantic similarity based approach outperforms the baseline Lesk algorithm by at least 15% on noun category and more than 100% on verb-category.

ACKNOWLEDGMENT

This work is partially supported by EU Marie Skodowska-Curie grant No 645706 and EU grant 770469-Cutler.

REFERENCES

[1] R. Navigli, Word sense disambiguation. A survey. ACM Computing Surveys 41(2), 2009, 10-69

[2] T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. Scientic American, 284(5): 28-37, 2001.

[3] M. Lesk, Automatic sense disambiguation using machine readable dictionaries: how to tell a pine code from an ice cream cone, in: Proceedings of the 5th annual international conference on Systems documentation, ACM Press, 1986, p. 24–26.

[4] S. Banerjee, T. Pedersen, An adapted Lesk algorithm for word sense disambiguation using WordNet, in: Proceedings of the Third International Conference on Intelligent Text Processing and Computational Linguistics, Mexico City, 2002, pp. 136–145.

[5] C. Fellbaum. WordNet – An Electronic Lexical Database, 1998, MIT Press.

[6] E. Agirre and D. Martinez. Decision Lists for English and Basque. Proceedings of the SENSEVAL-2 Workshop. In conjunction with ACL'2001/EACL'2001. Toulouse, France.

[7] R. Mihalcea and D. I. Moldovan. eXtended WordNet: Progress report. In Proceedings of NAACL Workshop on WordNet and Other Lexical Resources, p 95-100, Pittsburg, PA, 2001.

[8] E. Agirre, G. Rigau, Word sense disambiguation using conceptual density, in: Proceedings of the 16th International Conference on Computational Linguistics, Copenhagen, 1996, pp. 16–22.

[9] N. Habash and B. Dorr, "A categorial variation database for English," in Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1, 2003, pp. 17-23.

[10] M. Muhidin, and M. Oussalah. A Comparative Study of Conversion Aided Methods for WordNet Sentence Textual Similarity. COLING 2014 (2014): 37-41.

[11] Z. Wu and M. Palmer, Verb semantics and lexical selection. In 32nd. Annual Meeting of the Association for Computational Linguistics, pages 133 –138, New Mexico State University, 1994, Las Cruces, New Mexico.

[12] R. Navigli and P. Velardi. Structural semantic interconnections: A knowledge based approach to word sense disambiguation. In Proceedings of Senseval-3, Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text, pages 179{182, Barcelona, Spain, 2004.

[13] B. Magnini and G. Cavagli_a. Integrating subject field codes into WordNet. In Proceedings of the Second International Conference on Language Resources and Evaluation (LREC 2000), p. 1413-1418, 2000.

[14] P. Edmonds and S. Cotton. Senseval-2: Overview. In Proceedings of the Second International Workshop on Evaluating Word Sense Disambiguation Systems, p 1-6, Toulouse, France, 2001.

[15] S. Atkins. Tools for computer-aided lexicography: The Hector project. Acta Linguistica Hungarica, 41:5-72, 1993.