A FUZZY BASED APPROACH FOR WORDSENSE DISAMBIGUATION USING MORPHOLOGICAL TRANSFORMATION AND DOMAIN LINK KNOWLDGE.

F. FAROOGHIAN
iHR, Aston University,
B4 7ET, Birmingham, UNITED KINGDOM

M. OUSSALAH, E. GILIAN
Centre for Ubiquitous Computing, Faculty of Information Technology and Electrical Engineering, University of Oulu. OULU, 90014- FINLAND

Abstract. This paper describes a fuzzy-based methodology in order to aggregate outcomes of distinct wordsense disambiguation algorithms. The latter are derived from standard Lesk algorithm, its WorldNet extension and new interpretations of the set-intersection that accounts for various WordNet domain knowledge and part-of-speech conversion. The fuzzy preference model imitates the fuzzy Borda voting scheme. The developed algorithms are evaluated according to SenseEval 2 competition dataset, where a clear improvement to the baseline algorithm has been testified.

Keywords: word sense disambiguation, fuzzy preference, semantic similarity, WordNet.

1 Introduction

Word sense ambiguity is inherent to human language and prevalent in all natural languages where a single 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 can be selected based on the context where it occurs [1].

The appropriate handling of word sense disambiguation (WSD) task can potentially provide a major breakthrough in the information retrieval systems where identification of correct sense of query terms yields a milestone breakthrough in document retrieval systems, question-answering systems, among others [2].

Simultaneous interest in the linguistic community to research the structure of corpus resulted in various types of manually annotated corpora that populated

senseval/Semeval evaluation[1] where several research competitions are held towards designing new algorithms for disambiguation task. Lesk algorithm [3], based on the amount of overlapping between the gloss of the target word (for each sense) and the glosses of the context words, and its various extensions, e.g., Banerjee and Pederson [4] where WordNet was used as a source of glosses, have often setup a standard in the field.

This paper contributes to the wordsense disambiguation effort in the following way. First, new enhanced Lesk-like algorithms are put forward using different interpretations of set-intersection. 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. Especially, 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 basis for subsequent semantic similarity. Second, a fuzzy preference based strategy is employed in order to aggregate the outcomes of various disambiguation algorithms. The performances of the suggested algorithms are evaluated using both Senseval-2 and SemCor datasets 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, while testing results are reported in Section 4.

2 Background and related work

A pioneer work in word sense disambiguation is the Lesk algorithm [3] where the particular sense of a target word corresponds to the sense whose gloss (definition of the sense) shares the largest number of words with glosses of the words in the phrase to be disambiguated.

More formally, let W_1, W_2 be two words whose meanings are defined in the dictionary $N_{W_1} = \{W_1^1, W_1^2, .. W_1^{n_1}\}$ and $N_{W_2} = \{W_2^1, W_2^2, .. W_2^{n_2}\}$, repectively. The W_1, W_2 are therefore assigned senses W_1^i and W_2^j, $i \in \{1,2,...,n_1\}$, $j \in \{1,2,...,n_2\}$ (in the context of a phrase containing W_1 and W_2) such that

$$\left|W_1^i \cap W_2^j\right| = \max_{1\le k\le n_1, 1\le l\le n_2} \left|W_1^k \cap W_2^l\right| \quad (1)$$

Typically, N_{W_1} and N_{W_2} rely on glosses found in traditional dictionaries, e.g., Oxford English dictionary.

The complexity search for (1) increases exponentially with the number of words in the phrase. A known approximated solution to this problem, referred to as a simplified Lesk algorithm consists of restricting the overlapping operation, for

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

each sense of the target word, to words surrounding the target word. More formally, using the above notations and given a sentence /phrase where two words W_1 and W_2 co-occurs, say

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

W_1, W_2 are assigned senses W_1^i and W_2^j, respectively, such that

$$\left|W_1^i \cap W_2 \cap W_3 \cap \ldots \cap W_m\right| = \max_k \left|W_1^k \cap W_2 \cap W_3 \cap \ldots \cap W_m\right| \quad (2)$$

$$\left|W_2^j \cap W_1 \cap W_3 \cap \ldots \cap W_m\right| = \max_k \left|W_2^k \cap W_1 \cap W_3 \cap \ldots \cap W_m\right| \quad (3)$$

With the emergence of lexical databases, especially, WordNet where word senses are grouped into synsets organized in a hierarchical organization that creates semantic relations, 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 (1); namely, W_1^i, W_2^j would stand for all wording involved in describing i[th] and j[th] synset of word W_1 and W_2, respectively. Agirre and Martinez [6] proposed to use a (WordNet-based) semantic similarity in order to identify the correct sense. The latter corresponds to the senses that maximize the semantic similarity of the two words as in (4).

$$(W_1, W_2) \rightarrow (W_1^i, W_1^j) : Sim(W_1^i, W_1^j) = \max_{k,l} Sim(W_1^k, W_1^l) \quad (4)$$

This approach works only if the two words belong to the same part-of-speech. Mihalcea and Moldovan [7] extended this concept to pairs of different part-of-speech, especially for noun-verb connected via syntactic relations such as verb-object, 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.

3 Method

3.1 Part-of-speech category conversion

In order to deal with the discrepancy of semantic information available for distinct part-of-speech where noun category has much richer hierarchy structure than other categories in WordNet lexical database, our approach consists of using a morphological transformation in order to transform all non-noun entities (identified through an initial part-of-speech tagging) into their corresponding noun entities. For this purpose, we used the Categorial Variation Database (CatVar) [9-10]. The PoS conversion augmented with CatVar is accomplished by finding the database cluster containing the word to be converted and replacing it with the 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. Others words whose entry cannot be found in WordNet are left unchanged, e.g., named-entities.

In the same spirit as Banerjee and Pedersen [4], the process of word sense disambiguation of an individual word, say W with senses $W^1,...,W^n$, in the context constituted of a sentence $S = \{W_1, W_2, ..., W_m\}$, where each component W_i can be assigned sense in $\{W_i^1, W_i^2, ..., W_i^{n_i}\}$ (n_i stands for the number of senses (synsets) of word W_i), involves the following. First, translating the non-noun senses into noun-sense using the aforementioned CatVar transformation, yielding for each W_i, $\{N_i^1, N_i^2, ..., N_i^{n_i}\}$. Second, calculating, for each sense W^k (k=1,n) of target word W, its associated score:

$$Score(W^k) = \sum_i \max_j Sim(N^k, N_i^j) \quad \textbf{(5)}$$

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

$$W^{k*} = \arg\max_{W^k} \max_k Score(W^k) \quad (6)$$

Inspired by SSI (structural semantic interconnection) algorithm [12], the implementation of (5-6) can be rendered simple using an iterative process by first selecting words S' in S that are monosemous, say:

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

So that the counterpart of (5) becomes

$$Score(W^k) = \sum_{j:W_j \in S'} Sim(N^k, N_j) \quad (7)$$

Expression (5-6) 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 WorldNet similarity measure with all words of the context sentence S.

3.2 Use of WordNet domain category

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

[2] http://wndomains.fbk.eu/

$D_i^j = \left\{ d_{ij}^1, d_{ij}^2, .., d_{ij}^{l_{ij}} \right\}$, j=1 to n_i i=1 to *m*, be the set of domain links associated to synset W_i^j. Similarly, let $D_0^k = \left\{ d_k^1, d_k^2, .., d_k^{p_k} \right\}$, 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 (5-6) is

$$Score\_d(W^k) = \left| \bigcap_{i=0,m} \left(\bigcap_j D_i^j \right) \right| \tag{8}$$

Therefore, the sense k* is chosen so that

$$W^{k*} = \arg\max_{W^k} \max_k Score\_d(W^k) \tag{9}$$

In case where all cardinalities |. | in (8) vanish 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:

$$Score\_d'(W^k) = \min_{j,k} \sum_{i=0,m-1} dist(D_i^j, D_{i+1}^k) \tag{10}$$

Therefore, the associated sense is determined as:

$$W^{k*} = \arg\min_{W^k} \min_k Score\_d'(W^k) \tag{11}$$

Especially, (10-11) expressions are triggered only if expression (8) 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 concerning, one should notice that most of synsets are rather assigned one single domain link, therefore, the hierarchical distance based scoring function (10-11) is the most applied one in the subsequent reasoning.

3.3 Fuzzy Borda voting scheme

In the classical Borda count each expert gives a mark to each alternative, according to the number of alternatives worse than it. The fuzzy variant [16] is a natural extension that allows the experts to show numerically how much some alternatives are preferred to the others, evaluating their preference intensities from 0 to 1. More specifically, let R^1, $R^2, .., R^m$ be the fuzzy preference relations of m experts over n alternatives, say, $x_1,..,x_n$, yielding a preference matrix intensity for each expert k: $\left[r_{ij}^k \right]_{i,j=1,n}$, where $r_{ij}^k = \mu_{R^k}(x_i, x_j)$ being the membership function of R^k, quantifying the degree of confidence in which the k-expert prefers alternative x_i to alternative x_j. The score assigned for k-th expert to alternative x_i is aggregated as:

$$r_k(x_i) = \sum_{j=1,n \ \& \ r_{i,j}^k > 0.5} r_{i,j}^k \tag{12}$$

Taking into account the score of each individual expert, the overall score of a given alternative x_i will be:

$$r(x_i) = \sum_{k=1}^{m} r_k(x_i) \tag{13}$$

On the other hand, a practical eliciting of the individual (fuzzy) preference from individual expert estimation w_i of the quantity of interest as suggested in [16]:

$$r_{i,j}^{k} = \frac{w_i}{w_i + w_j} \tag{14}$$

Application to disambiguation

The key in applying fuzzy Borda voting scheme to the aforementioned problem of wordsense disambiguation is first to assume the aforementioned methodologies for wordsense disambiguation as an expert in the sense of Borda voting scheme. More specifically, we shall consider four distinct experts corresponding to following:

R^1: Lesk-WordNet as in expression (4); R^2: Lesk-WordNet-CatVar as in (5)

R^3: Lesk-WordNet-Monosemous as in (7); R^4:Lesk- domain category as in (8,10)

Second, the various alternatives x_i, correspond to the various senses of the target word to be disambiguated. Third, the estimation score yielded by each of the above disambiguation method R^i with respect to specific sense will be used through (14) to elicit the membership grade $r_{i,j}^{k}$ Fourth, the outcome of the voting scheme corresponds to the sense x_j that yields the highest score in the sense of (13).

4 Evaluation

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. The results in terms of precision, recall and runtime are reported in Table 1, together with comparison with some of the state of art approaches. The Lesk's algorithm is taken as a baseline for this analysis.

The results in Table 1 demonstrate 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 by more than 19% in both precision and recall evaluations. Among the four algorithms introduced in this paper, the CatVar-semantic similarity shows a marginal improvement over the use of domain category, monosemous and WordNet based Lesk's extension. On the other hand,

the use of fuzzy Borda voting scheme is also shown to improve, although, sometimes marginally the precision and recall performances with respect to the individual disambiguation algorithms (R^i, i=1,4).

Table 1. Classification results of the developed disambiguation algorithms on SenseEval 2 competition dataset

Algorithm	Noun(%)		Verbs (%)		Runtime (s)
	P	R	P	R	
R^1	71	68	58	55	.071
R^2	73	69	60	54	.082
R^3	68	59	54	53	.051
R^4	73	66	62	53	.042
Lesk (baseline)	61	60	21	18	.0049
Simple Lesk	32	28	29	28	.0143
Adapted Lesk	34	29	23	27	.0182
Fuzzy Borda Voting	74	69	69	58	.0983

Conclusion

This paper contributes to the hot topic of wordsense disambiguation and four new extensions of standard Lesk's algorithm have been provided based on the use of CatVar morphological transformation, domain link and monosemous. Next, a fuzzy-Borda voting scheme has been adapted in order to combine the outcomes of the various individual disambiguation algorithms to provide a global result. Although, the results are promising, the study prompted several interesting issues that will be further enhanced. For instance, the individual disambiguation algorithms are not fully independent from each other, which triggers interesting scenarios of accounting for the dependency level in the fuzzy Borda voting scheme. On the other hand, further theoretical results and convergence properties are still required in order to guarantee the superiority of the voting outcome over individual expert assessment.

Acknowledgment

This work is (partially) funded by the Marie Skłodowska-Curie Actions (645706-GRAGE)

References

1. R. Navigli, ACM Comp. Surv.41(2), 10 (2009)
2. T. Berners-Lee, J. Hendler, and O. Lassila. Sci. Amer. 284(5) 28 (2001).

3. M. Lesk, Proc. 5th ann. Int. Conf. Syst. Docu. ACM Press, 24 (1986).
4. S. Banerjee, T. Pedersen, Proc. 3rd Int. Conf. Intel. Text Proc. Comp. Ling. 136 (2002).
5. C. Fellbaum. WordNet – An Electronic Lexical Database, MIT Press, 1988.
6. E. Agirre and D. Martine, Proc. SENSEVAL-2 Work. ACL'2001/EACL'2001.
7. R. Mihalcea and D. I. Moldovan, Proc. NAACL WordNet and Other Lex. Res. 95 (2001).
8. E. Agirre, G. Rigau, Proc. 16th Int. Conf. Comp. Ling. 16 (1996).
9. N. Habash and B. Dorr, *Proc. North Amer. Chap. Assoc. Comp. Ling. Hum. Lang. Tech. 1*, 17, (2003).
10. M. Muhidin, and M. Oussalah. Proc. *COLING'14 37* (2014).
11. Z. Wu and M. Palmer, Proc. 32nd, Ann. Meet. Assoc. Comp. Ling. 133 (1994).
12. R. Navigli and P. Velardi, Proc. 3rd Int. Work. Eval. of Syst. Sem. Anal. Text, 179 (2004).
13. B. Magnini and G. Cavagli, Proc. 2nd Int. Conf. Lang. Res. Eval. (LREC 2000), 1413 (2000).
14. P. Edmonds and S. Cotton. Proc. 2nd Int. Work. Eval. Word Sense Disam. Syst. 1 (2001).
15. S. Atkins. Tools for computer-aided lexicography: The Hector project. Acta Ling. Hung. 41, 5 (1993).
16. H. Nurmi, *Group, Dec. Negot.* 10(2), 177 (2001)