Word Sense Disambiguation for Exploiting Hierarchical Thesauri in Text Classification

Dimitrios Mavroeidis[1], George Tsatsaronis[1], Michalis Vazirgiannis[1], Martin Theobald[2], and Gerhard Weikum[2]

[1] Department of Informatics, Athens University of Economics and Business, Greece
[2] Max-Planck Institute of Computer Science,
Saarbruecken, Germany

Abstract. The introduction of hierarchical thesauri (HT) that contain significant semantic information, has led researchers to investigate their potential for improving performance of the text classification task, extending the traditional "bag of words" representation, incorporating syntactic and semantic relationships among words. In this paper we address this problem by proposing a Word Sense Disambiguation (WSD) approach based on the intuition that word proximity in the document implies proximity also in the HT graph. We argue that the high precision exhibited by our WSD algorithm in various humanly-disambiguated benchmark datasets, is appropriate for the classification task. Moreover, we define a semantic kernel, based on the general concept of GVSM kernels, that captures the semantic relations contained in the hierarchical thesaurus. Finally, we conduct experiments using various corpora achieving a systematic improvement in classification accuracy using the SVM algorithm, especially when the training set is small.

1 Introduction

It can be argued that WSD algorithms for the document classification task should differ in their design and evaluation from pure WSD algorithms. It is expected that correctly disambiguated words could improve (and certainly not degrade) the performance of a document classification task, while falsely disambiguated words would entail noise. Although the SVM algorithm [1] used in our experiments is known to be noise tolerant, it is certain that noise, above a certain level, will eventually degrade in SVM's performance. In the absence of theoretical or experimental studies on the exact level of falsely disambiguated words that can be tolerated by classification algorithms, the most appropriate performance measure for WSD algorithms designed for a classification task is precision. Choosing the WSD algorithm with the highest precision will result in the incorporation of the lowest amount of noise in the classification task.

Another important issue for the successful embedding of WSD in text classification, is the exploitation of senses' semantic relations, that are provided by the HT. These relations are essential for defining distances and kernels that reflect semantic similarities between senses. An extensive bibliography exists for measuring distances and similarities on thesauri and ontologies, which has not been taken into account by other research

approaches that embed WSD in the text classification task. The need for exploiting semantic relations is illustrated in [2], where SemCor 1.7.1, a humanly-disambiguated corpus, is used in classification experiments. It is demonstrated that even with a 100% accurate disambiguation, the simple use of senses instead of keywords does not improve classification performance.

In this paper we propose an unsupervised WSD algorithm for classification, that utilizes a background HT. Our approach adopts the intuition that adjacent terms extracted from a given document are expected to be semantically close to each other and that is reflected to their pathwise distance on the HT. Thus, the objective of our WSD method is, given a set of terms, to select the senses (one for each term among many found in the HT) that overall minimize the pathwise distance and reflect the compactness of the selected sense set. The semantic compactness measure introduced is based on the concept of the Steiner Tree [3]. As opposed to other approaches that have utilized WSD for classification [4],[5],[6],[7], we have conducted extensive experiments with disambiguated corpora (Senseval 2 and 3, SemCor 1.7.1), in order to validate the appropriateness of our WSD algorithm. Experiments, using the WordNet HT, demonstrate that our WSD algorithm can be configured to exhibit very high precision, and thus can be considered appropriate for classification. In order to exploit the semantic relations inherent in the HT, we define a semantic kernel based on the general concept of GVSM kernels [8]. Finally, we have conducted experiments utilizing various sizes of training sets for the two largest Reuters-21578 categories and a corpus constructed from crawling editorial reviews of books from the Amazon website. The results demonstrate that our approach for exploiting hierarchical thesauri semantic information contributes significantly to the SVM classifier performance, especially when the training set size is small.

In the context of this paper WordNet [9] is utilized as a hierarchical thesaurus both for WSD and for classification. Although WordNet contains various semantic relations between concepts[1], our approach relies only on the hypernym/hyponym relation that orders concepts according to generality, and thus our approach can generalize to any HT that supports the hypernym/hyponym relation.

The rest of the paper is organized as follows. Section 2 discusses the preliminary notions and the related work. Section 3 presents our compactness measure for WSD that is based on the graph structure of an HT. Section 4 describes the semantic kernel that is utilized for the experiments. Section 5 discusses the experiments performed. Section 6 contains the comparison of the proposed framework to other approaches, concluding remarks and pointers to further work.

2   Preliminaries

2.1   Graph Theoretic Notions

Assuming that a document is represented by a set of senses, the semantic compactness measure that we introduce for WSD implies a similarity notion either among the senses of a sense set or between two sense sets. Its commutation is based on the notion of

---

[1] Concepts are word senses in WordNet terminology and in this paper we will use the terms word senses and concepts interchangeably.

Steiner Tree. Given a set of graph vertices, the Steiner Tree is the smallest tree that connects the set of nodes in the graph. The formal definition of the Steiner Tree is given below.

Definition 1 (Steiner Tree). *Given an undirected graph $G = (V, E)$, and a set $S \subseteq V$, then the Steiner Tree is the minimal Tree of G that contains all vertices of S.*

2.2 Semantic Kernels Based on Hierarchical Thesaurus

Since we aim at embedding WSD in the SVM classifier, we require the definition of a kernel that captures the semantic relations provided by the HT. To the extend of our knowledge the only approach that defines a semantic kernel based on a HT is [10]. The formal definition of their kernel is given below.

Definition 2 (Semantic Smoothing Kernels [10]). *The Semantic smoothing Kernel between two documents d_1, d_2 is defined as $K(d_1, d_2) = d_1 P' P d_2 = d_1 P^2 d_2$, where P is a matrix whose entries $P_{ij} = P_{ji}$, represent the semantic proximity between concepts i and j.*

The similarity matrix P is considered to be derived by a HT similarity measure. The Semantic Smoothing Kernels have similar semantics to the GVSM model defined in [8]. A kernel definition based on the GVSM model is given below.

Definition 3 (GVSM Kernel). *The GVSM kernel between two documents d_1 and d_2 is defined as $K(d_1, d_2) = d_1 D D' d_2$, where D is the term document matrix.*

The rows of matrix D, in the GVSM kernel contain the vector representation of terms, used to measure their pairwise semantic relatedness. The Semantic Smoothing Kernel has similar semantics. The Semantic Smoothing Kernel between two documents $K(d_1, d_2) = d_1 P^2 d_2$, can be regarded as a GVSM kernel, where the matrix D is derived by the decomposition of $P^2 = DD'$ (the decomposition is always possible since P^2 is guaranteed to be positive definite). The rows of D can be considered as the vector representation of concepts, used to measure their semantic proximity. Semantic Smoothing Kernels use P^2 and not P, because P is not guaranteed to be positive definite.

2.3 Related Work

WSD. The WordNet HT has been used for many supervised and unsupervised WSD algorithms. In direct comparison to our WSD approach we can find [11],[12],[13] that are unsupervised and rely on the semantic relations provided by WordNet. In the experimental section we show that our WSD algorithm can be configured to exhibit very high precision in various humanly-disambiguated benchmark corpora, and thus is more appropriate for the classification task.

Senseval (*www.senseval.org*), provides a forum, where the state of the art WSD systems are evaluated against disambiguated datasets. In the experimental sections we will compare our approach to the state of the art systems that have been submitted to the Senseval contests.

WSD and classification. In this section we shall briefly describe the relevant work done in embedding WSD in the document classification task. In [7], a WSD algorithm based on the general concept of Extended Gloss Overlaps is used and classification is performed with an SVM classifier for the two largest categories of the Reuters-25178 collection and two IMDB movie genres (*www.imdb.com*).

It is demonstrated that, when the training set is small, the use of WordNet senses together with words improves the performance of the SVM classification algorithm, however for training sets above a certain size, the approach is shown to have inferior performance to term-based classification. Moreover, the semantic relations inherent in WordNet are not exploited in the classification process. Although the WSD algorithm that is employed is not verified experimentally, its precision is estimated with a reference to [13], since the later work has a very similar theoretical basis. The experiments conducted by [13] in Senseval 2 lexical sample data, show that the algorithm exhibits low precision (around 45%) and thus may result in the introduction of much noise that can jeopardize the performance of a classification task.

In [4], the authors experiment with various settings for mapping words to senses (no disambiguation, most frequent sense as provided by WordNet and WSD based on context). Their approach is evaluated on the Reuters-25178, the OSHUMED and the FAODOC corpus, providing positive results. Their WSD algorithm has similar semantics to the WSD algorithm proposed in [12]. Although in [12] the experiments are conducted in a very restricted subset of SemCor 1.7.1, the results reported can be compared with our experiment results for the same task, as it is shown in Section 5. Moreover [4], use hypernyms for expanding the feature space.

In [5] the authors utilize the supervised WSD algorithm proposed in [14] in k-NN classification of the 20-newsgroups dataset. The WSD algorithm they employ is based on a Hidden Markov Model and is evaluated against Senseval 2, using "English all words task", reporting a maximum precision of around 60%. On the classification task of the 20-newsgroup dataset, they report a very slight improvement in the error-percentage of the classification algorithm. The semantic relations that are contained in WordNet are not exploited in the k-NN classification process.

The authors in [6] present an early attempt to incorporate semantics by means of a hierarchical thesauri in the classification process, reporting negative results on the Reuters-21578 and DigiTrad collection. While none disambiguation algorithm is employed, the use of hypernyms for extending the feature space representation is levied.

2.4   Hierarchical Thesaurus Distances – Similarities

As we have discussed in the introduction section, an important element for the successful incorporation of semantics in the classification process is the exploitation of the vast amount of semantic relations that are contained in the HT. There is an extensive bibliography that addresses the issue of defining distances and similarity measures based on the semantic relations provided by an HT [9],[15],[16],[17], which has not been related to the existing approaches for embedding WSD in classification. A common ground of most of the approaches is that the distance or similarity measure will depend on the "size" of the shortest path that connects the two concepts through a common ancestor in the hierarchy, or on the largest "depth" of a common ancestor in the hierarchy. The

terms "size" and "depth" are used in an informal manner, for details one should use the references provided.

3   Compactness Based Disambiguation

In this section we present our unsupervised WSD method, as this was initially sketched in [18]. Our WSD algorithm is based on the intuition that adjacent terms extracted from a text document are expected to be semantically close to each other. Given a set of adjacent terms, our disambiguation algorithm will consider all the candidate sets of senses and output the set of senses that exhibits the highest level of semantic relatedness. Therefore, the main component of our WSD algorithm is the definition of a semantic compactness measure for sets of senses. We refer to our disambiguation approach as CoBD (Compactness Based Disambiguation). The compactness measure utilized in CoBD is defined below.

Definition 4. *Given an HT O and a set of senses $S = (s_1, ..., s_n)$, where $s_i \in O$ the compactness of S is defined as the cost of the Steiner Tree of $S \cup lca(S)$, such that there exists at least one path, using hypernym relation, from each s_i to the $lca(S)$.*

In the definition above we include one path, using the hypernym relation, for every sense to the least common ancestor $lca(S)$. The reason for imposing such a restriction is that the distance between two concepts in an HT is not defined as the shortest path that connects them in the HT, but rather as the shortest path that goes through a common ancestor. Thus, it can be argued that two concepts are connected only through a common ancestor and not through any other path in the HT. The existence of the $lca(S)$ (and of a path between every concept and the $lca(S)$ using the hypernym relation) guarantees that a path connecting all pairs of concepts (in the context discussed earlier) exists.

Although in general the problem of computing the Steiner Tree is NP-complete, the computation of the Steiner Tree (with the restriction imposed) of a set of concepts with their lca in a HT is computationally feasible and is reduced to the computation of the shortest path of the lca to every concept of the set. Another issue, potentially adding excessive computational load, is the large number of combinations of possible sets of senses, when a term set of large cardinality is considered for disambiguation. In order to address this issue, we reduce the search space by using a Simulated Annealing algorithm. The experimental setup used in this paper for the empirical evaluation of our WSD algorithm is described in detail in section 5.

4   Exploitation of Hierarchical Thesaurus Semantics in SVM Classification

We have argued in the introductory section that the exploitation of the semantics provided by an HT are important for the successful embedding of WSD in the classification task. In this section we will present the definition of the Kernel we will utilize in SVM classification. The Kernel we define is based on the general concept of GVSM kernel and depicts the semantics of the HT.

It is shown in detail in [18], that the use of hypernyms for the vector space representation of the concepts of a HT, enables the measurement of semantic distances in the vector space. More precisely, given a Tree HT, there exists a weight configuration for the hypernyms, such that standard vector space distance and similarity measures are equivalent to popular HT distances and similarities. The proofs for propositions given below can be found in [18].

Proposition 1. *Let O be a Tree HT, if we represent the concepts of the HT O, as vectors containing all their hypernyms, then there exists a configuration for the weights of the hypernyms such that the Manhattan distance (Minkowski distance with p=1) of any two concepts in vector space is equal to the Jiang-Conrath measure [15] in the HT.*

Proposition 2. *Let O be a Tree HT, if we represent the concepts of the HT O as vectors containing all their hypernyms, then there exists a configuration for the weights of the hypernyms such that the inner product of any two concepts in vector space is equal to the Resnik similarity measure [16] in the HT.*

The WordNet hierarchical thesaurus is composed by 9 hierarchies that contain concepts that inherit from more than one concept, and thus are not Trees. However, since only 2.28% of the concepts inherit from more than one concept [19], we can consider that the structure of WordNet hierarchies is close to the Tree structure.

From the above we conclude that, if we construct a matrix D where each row contains the vector representation of each sense containing all its hypernyms, the matrix DD' will reflect the semantic similarities that are contained in the HT. Based on D, we move on to define the kernel between two documents d_1, d_2, based on the general concept of GVSM kernels as $K_{concepts}(d_1, d_2) = d_1 DD' d_2$. In our experiments we have used various configurations for the rows of D. More precisely, we have considered the vector representation of each concept to be extended with a varying number of hypernyms. The argument for using only a limited number and not all hypernyms is that the similarity between hypernyms close to the root of the HT is considered to be very close to 0. Apart from hypernyms, in the experiments section, we have explored the potential of using hyponyms for constructing matrix the D in the GVSM kernel. The kernel that we finally utilize in our experiments is a combination of the inner product kernel for terms with the concept kernel $K(d_1, d_2) = K_{terms}(d_1, d_2) + K_{concepts}(d_1, d_2)$. This kernel was embedded into the current version of *SVMLight* [20] and replaced the standard linear kernel used for document classification with sparse training vectors.

The kernel defined implies a mapping from the original term and concept space, to a space that includes the terms, the concepts and their hypernyms. The kernel can be considered as the inner product in this feature space.

5   Experiments

5.1   Evaluation of the WSD Method

CoDB was tested in four benchmark WSD corpora; Brown 1 and Brown 2 from the SemCor 1.7.1 corpus, and the in the "English All Words" task of Senseval 2 and 3. These corpora are pre-tagged and pre-annotated. From all the parts of speech in the
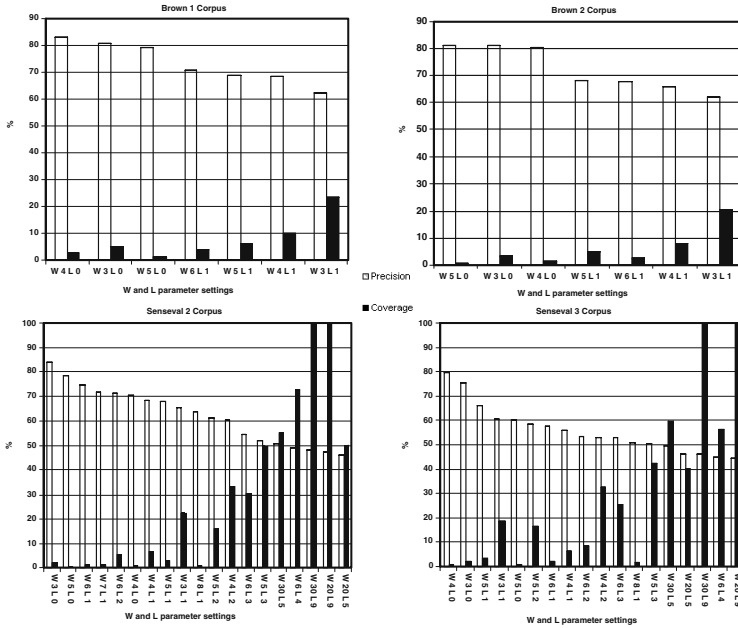
Fig. 1. WSD results on 4 benchmark datasets for different initializations of W and L

texts we only considered nouns, which are usually more informative than the rest and form a meaningful type hierarchy in WordNet. In order to implement CoBD efficiently we had to take into account that the search space of combinations to be examined for their compactness increases dramatically as the cardinality of the set of words examined increases, making exhaustive computation infeasible. Thus we adopted simulated annealing as in [21]. This approach reduced the search space and allowed us to execute the WSD using various set of words sizes in a time efficient manner. The parameters of the WSD method are:

1. Window Size (W): Set cardinality of the words to be disambiguated.
2. Allowed Lonely: Given a word set L, it is the maximum number of lonely senses [1] allowed in a WordNet noun hierarchy, for any senses combination of that window.

Figure 1 presents experiments we have conducted using various parameter settings. The results are sorted in decreasing order of precision. The precision and coverage [2] values reported do not take into account the monosemous nouns, but only the ambiguous ones. We can estimate, based on the examined corpora statistics, that the inclusion of the monosemous nouns would report an increase in precision between 3% and 4%, as well as an increase in coverage of almost 22%.

---

[1] A sense s belonging to a set of senses S is referred to as lonely if the WordNet noun hierarchy H it belongs to, does not contain any other $k \in S$.

[2] Coverage is defined as the percentage of the nouns that are disambiguated.

We observe that CoBD achieves precision greater than 80% with an associated coverage of more than 25%, if monosemous (i.e., non-ambiguous) nouns are also taken into account. Comparable experiments conducted in [12] reported a top precision result of 64,5% with an associated coverage of 86,2%. Similar experiments conducted in [11], [13] and [14] resulted as well in lower precision than CoBD. In comparing our approach to the state of the art WSD algorithms that were submitted to the "English All Words" Senseval 2 contest (*www.senseval.org*), we observe that our approach can be configured to exhibit the highest precision.

5.2   Document Collections and Preprocessing for Text Classification

Reuters. Reuters-21578 is a compilation of news articles from the Reuters newswire in 1987. We include this collection mostly for transparency reasons, since it has become the gold standard in document classification experiments. We conducted experiments on the two largest categories, namely *acquisitions* and *earnings*, in terms of using test- and training documents based on the [4] split. This split yields a total of 4,436 training and 1,779 test documents for the two categories. We extracted features from the mere article bodies, thus using whole sentences only and hiding any direct hint to the actual topic from the classifier.

Amazon. To test our methods on a collection with a richer vocabulary, we also extracted a real-life collection of natural-language text from amazon.com using Amazon's publicly available Web Service interface. From that taxonomy, we selected all the available editorial reviews for books in the three categories *Physics*, *Mathematics* and *Biological Sciences*, with a total of 6,167 documents. These reviews typically contain a brief discussion of a book's content and its rating. Since there is a high overlap among these topics' vocabulary and a higher diversity of terms within each topic than in Reuters, we expect this task to be more challenging for both the text- as well as the concept-aware classifier.

Before actually parsing the documents, we POS-annotated both the Reuters and Amazon collections, using a version of the commercial Connexor software for NLP processing. We restricted the disambiguation step to matching noun phrases in Word-Net, because only noun phrases form a sufficiently meaningful HT in the ontology DAG. Since WordNet also contains the POS information for each of its concepts, POS document tagging significantly reduces the amount of choices for ambiguous terms and simplifies the disambiguation step. For example the term *run* has 52 (!) distinct senses in WordNet out of which 41 are tagged as verbs. The parser first conducts continuous noun phrase tokens in a small window of up to a size of 5 into dictionary lookups in WordNet before the disambiguation step takes place. If no matching phrase is found within the current window, the window is moved one token ahead. This sliding window technique enables us to match any composite noun phrase known in WordNet, whereupon larger phrases are typically less ambiguous. Non-ambiguous terms can be chosen directly as safe seeds for the compactness-based disambiguation step. Note that we did not perform any feature selection methods such as Mutual Information or Information Gain [22] prior to training the SVM, in order not to bias results toward a specific classification method.
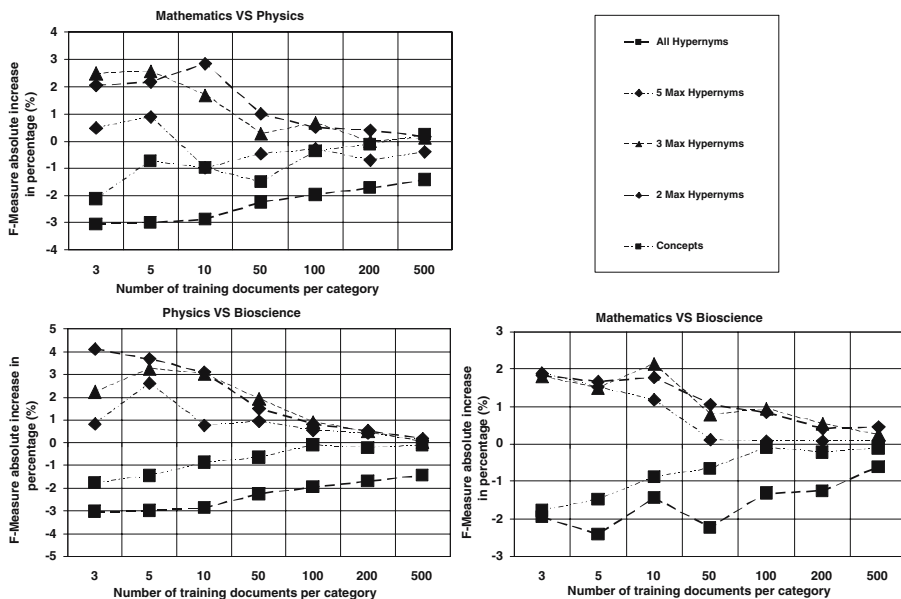
Fig. 2. Relative Improvement of F-measures scores for various Similarity Configurations in the Amazon Topics

5.3   Evaluation of Embedding CoBD in the Text Classification Task

To evaluate the embedding of CoBD in text classification, we performed binary classification tasks, only, i.e., we did not introduce any additional bias from mapping multi-class classification task onto the binary decision model used by the SVM method. The binary classification tasks were performed after forming all pairs between the three Amazon topics, and one pair between the two largest Reuters-21578 topics. The parameters' setting for CoBD was *W 3 L 0*, since it reported high percision and performed in a stable manner during the WSD evaluation experiments in the 4 benchmark corpora. Our baseline was the F-Measure [22] arising from the mere usage of term features. The baseline competed against the embedding of the term senses, whenever disambiguation was possible, and their hypernyms/hyponyms into the term feature vectors, according to the different GVSM kernel configurations shown in Figures 2,3. In our experiments, the weights of the hypernyms used in the GVSM kernel are taken to be equal to the weights of the terms they correspond to. We varied the training set sizes between 3 and 500 documents per topic. For each setup, in Figures 2,3 we report the differences of the *macro-averaged F-Measure* between the baseline and the respective configurations, using 10 iterations for each of the training set sizes of the Reuters dataset and 30 iterations for each of the training set sizes of the Amazon dataset. The variation of the differences was not too high and allowed for all the results where the absolute difference of the sample means was greater than 1% to reject the null hypothesis (that the means are equal) at a significance level of 0.05. For more than 500 documents, all our experiments
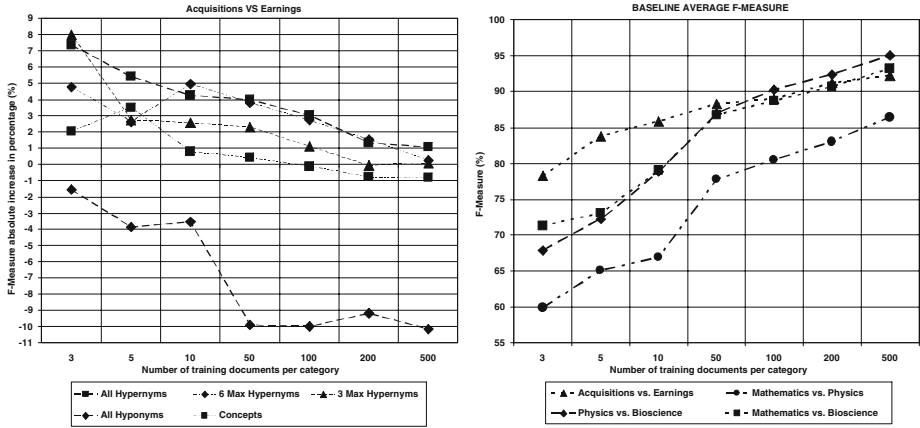
Fig. 3. Relative Improvement of F-measures scores for various Similarity Configurations in the Reuters Topics

indicate a convergence in results between the concept-aware classifier and the text classifier. The average F-measures for the baseline classifier are reported in Figure 3. For each run, the training documents were selected randomly following a uniform distribution. Since there is no split into separate documents for training and testing given in the Amazon collection, we performed cross-validation runs over the whole set, each using all the remaining documents for the test phase.

The results demonstrate that the use of CoBD and our kernel function, based on a small number of hypernyms increases consistently the classification quality especially for small training sets. In some cases, as the number of hypernyms increases we observe a performance deterioration which in some cases falls below the term-based classification.

The variance in the number of hypernyms needed for achieving better performance, can be explained by the fact that we did not employ a hypernym weighting scheme. Thus, when semantically correlated categories are considered, (such as Maths/Physics in the Amazon data), then the use of all the hypernyms with equal weights would result in many documents belonging to the Physics category to have a high similarity to documents of Maths category, degrading the performance of the classification algorithm.

6   Discussion and Conclusions

The context of the current work entails the content and structure (i.e. the senses and hierarchical relationships) of HTs and their usage for successful extension of the bag of words model for text classification. The objective is that such extensions (i.e. senses and hypenyms/hyponyms more precisely) are contributing to higher quality in the classification process.

The *contribution* of the paper is the design of a successful WSD approach to be incorporated and improve the text classification process. Our WSD approach takes into account term senses found in HTs, (in the specific case Wordnet), and for each document

selects the best combination of them based on their conceptual compactness in terms of related Steiner tree costs. Apart from the senses we add to the original document feature set a controlled number of hypernyms of the senses at hand. The hypernyms are incorporated by means of the kernel utilized. The attractive features of our work are:

Appropriate WSD approach for text classification. Most of the related approaches incorporating WSD in the classification task [6],[7],[4] do not provide a sound experimental evidence on the quality of their WSD approach. On the contrary in our work, the WSD algorithm is exhaustively evaluated against various humanly disambiguated benchmark datasets and achieves very high precision (among the top found in related work) although at low coverage values (see Fig.1). This is not a problem, though since as mentioned earlier, it is essential to extend the feature space with correct features in order to prevent introduction of noise in the classification process. The experimental evaluation provides us with the assurance that our WSD algorithm can be configured to have high precision, and thus, would insert in the training set very little noise.

Similarity measure that takes into account the structure of the HT. Document classification depends on a relevant similarity measure to classify a document into the closest of the available classes. It is obvious that the similarity among sets of features (representing documents) should take into account their hierarchical relationships as they are represented in the HT. None of the previous approaches for embedding WSD in classification has taken into account the existing literature for exploiting the HT relations. Even when the use of hypernyms is used [6],[4], it is done in an ad-hoc way, based on the argument that the expansion of a concept with hypernyms would behave similar to query expansion using more general concepts. We utilize a Kernel based on the general concept of a GVSM kernel that can be used for measuring the semantic similarity between two documents. The kernel is based on the use of hypernyms for the representation of concepts - theoretically justified in the context of the related work concerning the computation of semantic distances and similarities on a HT that aligns to tree structure.

We conducted classification experiments on two real world datasets (the two largest Reuters categories and a dataset constructed by the editorial reviews of products on three categories at the *amazon.com* web site). The results demonstrate that our approach for embedding WSD in classification yields significantly better results especially when the training sets are small.

An issue that we will investigate in further work is the introduction of a weighting scheme for hypernyms favoring hypernyms that are close to the concept. A successful weighting scheme is expected to reduce the problem of the variance in the number of hypernyms needed to achieve optimal performance. We will investigate learning approaches to learn the weighting schemes for hypernyms. Moreover, we aim in conducting further experiments on other larger scale and heterogeneous data sets.

References

1. Drucker, H., Burges, C.J.C., Kaufman, L., Smola, A.J., Vapnik, V.: Support vector regression machines. In: Advances in Neural Information Processing Systems (NIPS). (1996) 155–161
2. Kehagias, A., Petridis, V., Kaburlasos, V.G., Fragkou, P.: A comparison of word- and sense-based text categorization using several classification algorithms. Journal of Intelligent Information Systems **21** (2003) 227–247

3. Hwang, R., Richards, D., Winter, P.: The steiner tree problem. Annals of Discrete Mathematics **53** (1992)

4. Bloehdorn, S., Hotho, A.: Boosting for text classification with semantic features. In: Proc. of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Mining for and from the Semantic Web Workshop. (2004) 70–87

5. Rosso, P., Ferretti, E., Jimenez, D., Vidal, V.: Text categorization and information retrieval using wordnet senses. In: Proc. of the 2nd International WordNet Conference (GWC). (2004)

6. Scott, S., Matwin, S.: Feature engineering for text classification. In: Proc. of the 16th International Conference on Machine Learning (ICML). (1999) 379–388

7. Theobald, M., Schenkel, R., Weikum, G.: Exploiting structure, annotation, and ontological knowledge for automatic classification of xml data. In: International Workshop on Web and Databases (WebDB). (2003) 1–6

8. Wong, S.K.M., Ziarko, W., Wong, P.C.N.: Generalized vector space model in information retrieval. In: Proc. of the 8th annual international ACM SIGIR conference on Research and development in information retrieval. (1985) 18–25

9. Fellbaum, C., ed.: WordNet, An Electronic Lexical Database. The MIT Press (1998)

10. Siolas, G., d'Alche Buc, F.: Support vector machines based on semantic kernel for text categorization. In: Proc. of the International Joint Conference on Neural Networks (IJCNN). Volume 5., IEEE Press (2000) 205–209

11. Sussna, M.: Word sense disambiguation for free-text indexing using a massive semantic network. In: Proc. of the 2nd International Conference on Information and Knowledge Management (CIKM). (1993) 67–74

12. Agirre, E., Rigau, G.: A proposal for word sense disambiguation using conceptual distance. In: Proc. of Recent Advances in NLP (RANLP). (1995) 258–264

13. Banerjee, S., Pedersen, T.: Extended gloss overlaps as a measure of semantic relatedness. In: Proc. of the 18th International Joint Conference on Artificial Intelligence (IJCAI). (2003) 805–810

14. Molina, A., Pla, F., Segarra, E.: A hidden markov model approach to word sense disambiguation. In: Proc. of the 8th Iberoamerican Conference on Artificial Intelligence. (2002)

15. Jiang, J., Conrath, D.: Semantic similarity based on corpus statistics and lexical taxonomy. In: Proc. of the International Conference on Research in Computational Linguistics. (1997)

16. Resnik, P.: Using information content to evaluate semantic similarity in a taxonomy. In: Proc. of the 14th International Joint Conference on Artificial Intelligence (IJCAI). (1995)

17. Lin, D.: An information-theoretic definition of similarity. In: Proc. of the 15th International Conference on Machine Learning (ICML). (1998) 296–304

18. Mavroeidis, D., Tsatsaronis, G., Vazirgiannis, M.: Semantic distances for sets of senses and applications in word sense disambiguation. In: Proc. of the 3rd International Workshop on Text Mining and its Applications. (2004)

19. Devitt, A., Vogel, C.: The topology of wordnet: Some metrics. In: Proc. of the 2nd International WordNet Conference (GWC). (2004) 106–111

20. Klinkenberg, R., Joachims, T.: Detecting concept drift with support vector machines. In: Proc. of the 17th International Conference on Machine Learning (ICML). (2000) 487–494

21. Cowie, J., Guthrie, J., Guthrie, L.: Lexical disambiguation using simulated annealing. In: 14th International Conference on Computational Linguistics (COLING). (1992) 359–365

22. Manning, C., Schuetze, H.: Foundations of Statistical Natural Language Processing. MIT Press (2000)