Procesamiento del Lenguaje Natural, Revista nº 49 septiembre de 2012, pp 49-56 recibido 04-05-12 revisado 04-06-12 aceptado 07-06-12

A clustering-based Approach for Unsupervised Word Sense Disambiguation

Una aproximación no supervisada para la desambiguación del sentido de las palabras basada en agrupamiento

Tamara Martín-Wanton
Department of Languages and Computer Systems, Universidad de Educación a Distancia (UNED)
C/ Juan del Rosal n 16. 28040 - Madrid
tmartin@lsi.uned.es

Rafael Berlanga-Llavori
Department of Languages and Computer Systems, Universidad Jaume I,
Ave. Vicent Sos Baynat, Castellón 12071
Spain
berlanga@lsi.uji.es

Resumen: Los métodos de agrupamiento han sido ampliamente usados en muchas tareas de Procesamiento de la Información con el fin de capturar categorías de objetos desconocidos. Sin embargo, el agrupamiento ha sido poco utilizado como método para etiquetar sentidos en la Desambiguación del Sentido de las Palabras (WSD), es decir, como una forma de identificar grupos formados por sentidos de palabras semánticamente relacionados que pueden ser utilizados con éxito en el proceso de desambiguación. En este artículo presentamos un método de desambiguación no supervisado basado en el agrupamiento de sentidos de palabras que además es capaz de encontrar relaciones implícitas (no presentes en WordNet) entre los sentidos de las palabras de la oración. Investigamos en profundidad el rol del agrupamiento y su contribución al WSD. En los resultados experimentales se demuestra la utilidad del agrupamiento para la desambiguación no supervisada.
Palabras clave: Desambiguación del Sentido de las Palabras, Agrupamiento

Abstract: Clustering methods have been extensively used in many Information Processing tasks in order to capture unknown object categories. However, clustering has been scarcely used as a sense labeling method for Word Sense Disambiguation (WSD), that is, as a way to identify groups of semantically related word senses that can be successfully used in a disambiguation process. In this paper, we present an unsupervised disambiguation method relying on word sense clustering that also reveals the implicit relationships (not asserted in WordNet) existing among these word senses.We also investigate in depth the role of clustering and its contribution to WSD. Experimental results demonstrate the usefulness of clustering for unsupervised WSD.
Keywords: Word Sense Disambiguation, Clustering

1. Introduction

The task of Word Sense Disambiguation (WSD) consists of assigning the appropriate meaning (sense) for a particular contextual occurrence of a polysemous word. This task is an essential research area in Natural Language Processing that contributes to almost all semantic-based text processing applications (e.g., Machine Translation, Information Extraction, Question & Answering, etc.).

Navigli (2009) broadly divides WSD approaches into supervised and unsupervised WSD. The former ones require learning a model from hand-tagged samples to disambiguate words, which give them a domain specific character. The latter ones are based on unlabeled corpora, avoiding thus the use of training samples. WSD approaches are further classified into knowledge-based and corpus-based methods. Knowledge-based methods exploit word relationships provided by external lexical resources (e.g., dictionaries, ontologies, etc.), whereas corpus-based methods do not make use of any of these resources. Currently, lexical resources like

ISSN 1135-5948 © 2012 Sociedad Española para el Procesamiento del Lenguaje Natural

WordNet (Miller, 1995) constitute the referred source in most general purpose approaches. In this paper, we focus on unsupervised and knowledge-based methods.

The main contribution of this paper is twofold. Firstly, we extend the knowledge-based framework proposed in (Anaya-Sánchez, Pons-Porrata, & Berlanga-Llavori, 2006) for the disambiguation of nouns and present an unsupervised all-words disambiguation method derived from it. Our proposal relies on word sense clustering as a natural way to capture the reflected cohesion among the words of a textual unit. This approach is also able to reveals the implicit relationships (not asserted in WordNet) existing among these word senses.

Secondly, we explore in depth the role of clustering and its contribution to WSD. Specifically, we evaluate the capability of clustering for identifying groups of semantically related senses that can help the selection of the right ones and compare our approach with the clustering scheme of senses induced by WordNet domains (Magnini & Cavaglià, 2000). Our experimental results demonstrate the usefulness of word sense clustering for unsupervised WSD.

2. Related work

Most of the knowledge-based methods can be broadly divided into two categories, namely, similarity- and graph-based ones (Navigli & Lapata, 2010). The first category compares each sense of a target word with its surrounding context words. The sense that has the highest similarity is assumed to be the right one. In these approaches, right senses are determined for each word individually without considering the senses previously assigned.

In graph-based methods (Mihalcea, 2005; Navigli & Lapata, 2010), a graph whose nodes are word senses and edges represent meaningful relations or dependencies between them, is built from lexical resources. This graph structure is assessed to determine the importance of each node and the right sense corresponds to the most important node for each word. Experimental studies (Mihalcea, 2005; Brody, Navigli, & Lapata, 2006) show that graph-based methods outperform similarity-based ones. Like Mihalcea's method, we build a weighted graph whose nodes are word senses and edges are labeled with the similarity between them, but instead of determining the importance of a sense by using centrality algorithms, we iteratively perform a clustering method to discover the relationships existing among senses to identify the right ones.

Clustering has been explicitly used in the WSD area for clustering textual contexts of words to induce word senses by dividing the word occurrences into a number of classes or senses (Pedersen, 2006), and also for clustering of fine-grained word senses into coarse-grained ones for reducing the polysemy degree of words (Agirre & López, 2003; Navigli, 2006). However, clustering has been scarcely used as a sense labeling method in the disambiguation task. Hence, our approach shows a novel way of using clustering in this field.

To the best of our knowledge, the major effort in providing groups of semantically related word senses for disambiguation purposes consists of the definition of WordNet domains and several disambiguation algorithms use this domain categories to improve the disambiguation results (Magnini et al., 2002; Kolte & Bhirud, 2008). However, as we demonstrate in the experiments section, the granularity level of these groups is too coarse to be useful for relating word senses.

Most of the WSD methods are restricted to determine the right sense of words in a given context, but none of them give additional information about the possible relationships among the disambiguated word senses. In our proposal, we attempt to reveal the implicit relationships (not asserted in WordNet) existing among word senses.

3. A knowledge-based framework for WSD

In this paper, we extend the framework firstly introduced in (Anaya-Sánchez et al., 2006) to all-words disambiguation. The underlying idea of the framework is to use clustering as a way of identifying semantically related word senses. The goal of the framework is the disambiguation of a finite set of words W given a textual context T. Here, we do not restrict the elements of W to be in T. Our framework comprises the following elements: (1) a representation for senses, which is provided by the knowledge source; (2) a similarity measure to compare sense representations; (3) a

clustering algorithm able to group the sense representations of all words in W; (4) a filtering function for selecting sense clusters that match the best with the context T, and (5) a stopping criterion for ensuring the termination of the disambiguation process.

Assuming that these elements are given, the disambiguation process of the framework starts from a clustering distribution of all possible senses of the words in *W*. Such a clustering tries to identify cohesive groups of word senses, which are assumed to represent different meanings for the set of words. Then, clusters that match the best with the context are selected via a filtering process. If the selected clusters disambiguate all words (i.e., they contain exactly one sense for each word in *W*), the process stops and the senses belonging to the selected clusters are interpreted as the right ones. Otherwise, the clustering and filtering steps are performed again (regarding the remaining senses) until the stopping condition is satisfied. It is worth mentioning that in each iteration the clustering parameters must be refined to obtain stronger cohesive clusters. Notice that in this framework word senses are globally determined by capturing relationships among senses via the clustering process. Figure 1 shows the general steps of the framework for the disambiguation of a set of words. See (Anaya-Sánchez et al., 2006) for details.

Input: The finite set of words *W* and the textual context *T*
Output: The disambiguated word senses
1. Let *Senses* be the set of all senses of words in *W*
2. Repeat
   a. G = *clustering*(*Senses*)
   b. *Selected_G* = *filter*(*G*, *W*, *T*)
   c. $Senses = \bigcup_{g \in Selected\_G} \{s \mid s \in g\}$
   until *stopping-criterion*
3. Return *Senses*

Figure 1: Framework for the disambiguation of the set of words *W* in the textual context *T*

4. Star-based Disambiguation Algorithm

In this section, we introduce our disambiguation approach, which derives from the framework explained above. Our algorithm proceeds incrementally on a sentence-by-sentence basis. We assume that sentences are part-of-speech tagged and, therefore only content words (i.e., nouns, verbs, adjectives, and adverbs) are considered. Thus, in our case the context *T* is represented as a vector of content words in a sentence, all weighted one, and *W* is the set of all words in *T*.

For example, let us consider again the sentence "*The runner won the marathon*". In this example, the set of words *W* includes the nouns *runner* and *marathon*, and the verb *win* (lemma of the verbal form *won*), and the context is the vector *T*=< *runner*:1, *win*:1, *marathon*:1>. The rest of words are not considered for they are meaningless (i.e., stopwords). Hereafter, we will use this sentence example to explain our approach.

4.1. Sense representation

For clustering purposes, word senses are represented as feature vectors. Thus, for each word sense *s* we define a vector $<t_1{:}\sigma_1, \ldots, t_m{:}\sigma_m>$, where each feature t_i is a WordNet term highly correlated to *s* with an association weight σ_i. The set of terms for a word sense includes all its WordNet hyponyms, its directly related terms (including coordinated terms) and their filtered and lemmatized glosses.

To weight vector terms, the *tf-idf* statistics is used, considering each word as a collection and its senses as the collection documents. It is worth mentioning that the use of *tf-idf* weights allows us to distinguish a sense from the other senses of the same word. In this paper, we use the cosine as similarity measure between sense representations:

$$cos(s_i, s_j) = \frac{\vec{s}_i \cdot \vec{s}_j}{\|\vec{s}_i\| \|\vec{s}_j\|}$$

4.2. Clustering algorithm

Sense clustering is carried out by the extended star clustering algorithm (Gil-García, Badía-Contelles, & Pons-Porrata, 2003), which builds star-shaped and overlapped clusters. This clustering algorithm relies on a greedy cover of the β_0-similarity graph by star-shaped subgraphs (Aslam, Pelekhov, & Rus, 2004). A β_0-similarity graph is the undirected graph, whose vertices are word senses and there is an edge between *sense* s_i and sense s_j if the similarity between them is greater than the minimum similarity threshold β_0. Each cluster (star-shaped subgraph) consists of a single star and its satellites, where the star is the word sense with the highest connectivity within the cluster, and the satellites are those senses

connected with the star (i.e., the star neighbors in the graph).

Notice that in our approach, the disambiguation is performed over all the senses of all words in the sentence at once. The underlying hypothesis is that word sense clustering captures the reflected cohesion among the words of a sentence and each cluster reveals possible relationships existing among these word senses. Thus, the way this clustering algorithm relates word senses resembles the way in which syntactic and discourse relations link textual elements.

4.3. Filtering

For each word *w* the proposed filter selects those clusters having maximum similarity w.r.t. the context *T*. That is, we use the following function:

$$filter(G,W,T) = \bigcup_{w \in W} \arg\max_{\substack{g \in G \\ g \text{ covers } w}} cos(\overline{g},T)$$

In this definition, $cos(\overline{g},T)$ represents the cosine similarity between the centroid of cluster *g* and the context *T*, and *covers* states the relation that links a cluster with those words having senses in it.

Thus, we firstly rank all clusters according to its similarity with the context *T*, and then they are orderly processed to select clusters for covering the words in *W*. A cluster *g* is selected if it contains at least one sense of an uncovered word. Otherwise it is discarded.

4.4. Stopping Criterion

As a result of the filtering process, a set of senses for each word in *W* is obtained (i.e., the union of all the selected clusters). Words in *W* having only one sense are considered disambiguated. If some word still remains ambiguous, we must refine the clustering process to get stronger cohesive clusters of senses. In this case, all the remaining senses must be clustered again but raising the β_0 threshold. Thus, this process must be done iteratively until either all words are disambiguated or β_0 cannot be increased anymore. Initially, β_0 is defined as:

$$\beta_0(1) = pth(p, cos(Senses))$$

and at the i-th iteration (i > 1) it is updated to:

$$\beta_0(i) = \min_{q \in \{5,10,15,\ldots\}} \left\{ \begin{array}{c} \beta = pth(p+q, cos(Senses)) \\ | \ \beta > \beta_0(i-1) \end{array} \right\}$$

In these equations, *Senses* is the set of current senses, and *pth*(*p, cos*(*Senses*)) represents the *p*-th percentile value of pair-wise cosine similarities of all senses in *Senses* (i.e., $cos(Senses)=\{cos(s_i, s_j) \mid s_i,s_j \in Senses, i \neq j\} \cup \{1\}$). Here, *p* is a user-defined parameter.

Figure 5 graphically depicts the disambiguation process of the example sentence carried out by our method. The boxes in the figure represent the obtained clusters, which are sorted regarding their similarities with respect to the context (scores are under the boxes), and doubly-boxed clusters depict the selected ones by the filter.

In our example, we select *p* = 90 for obtaining the initial similarity threshold (β_0=0.048). Notice that the first cluster includes senses that cover the set of all the ambiguous words. Hence, it is selected by the filtering process and all other clusters are discarded. After this step, *Senses* is updated with the senses of the selected cluster.

At this point of the process, *Senses* does not disambiguate completely *W* because the noun *runner* has still two senses. Consequently, a new clustering must be obtained using the current set *Senses* and a new value of β_0.

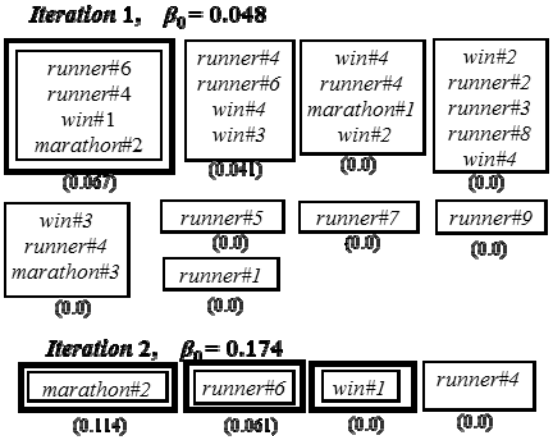

Figure 5: Disambiguation of words in "The runner won the marathon"

As *pth*(90+5, *cos*(*Senses*)) = 0.174 and 0.174 > 0.048, then $\beta_0(2)$ = 0.174. In this case, all clusters become single. Then, the final set of selected senses is *Senses*= {*marathon*#2, *runner*#6, *win*#1}, which includes only one sense for each word in *W*.

5. Experiments

In order to evaluate our proposal, we use the coarse-grained English all-words corpus of SemEval-2007 Task 07 (Navigli, Litkowski, &

Hargraves, 2007). This corpus consists of 5,377 words of running text of which 2,269 have been annotated with senses from a coarse-grained version of the WordNet 2.1 sense inventory.

We follow the evaluation methodology of SemEval-2007 and present the disambiguation results in terms of the traditional F1-measure.

5.1. Does the Extended Star Clustering Algorithm Produce Profitable Clusters for WSD?

The aim of the first experiment is to validate the performance of the extended star algorithm for clustering semantically related senses, i.e., for obtaining useful groups for WSD. With this purpose, for each corpus sentence we compare the relation between the senses generated from the clustering algorithm (namely, the set of pairs (u, v) such that senses u and v belong to a same cluster) w.r.t. the reference model consisting of all pairs of correct word senses.

We use *recall*, *precision* and F1 to evaluate these relations, which can be expressed as follows:

$$recall = \frac{c}{a} \qquad precision = \frac{c}{b} \qquad F1 = \frac{2c}{a+b}$$

where c is the number of pairs of correct senses generated from the clustering, a is the number of all pairs of correct senses in the reference model, and b is the number of sense pairs produced from the clustering that include at least one correct sense. Regarding that our reference model does not include any relation between incorrect senses, we discard all pairs of incorrect senses obtained from the clusters.

Notice that in the above definitions *recall* is a measure of the goodness on grouping together correct senses; whereas *precision* measures the accuracy of the clustering for relating correct word senses with themselves. The F1 measure is the harmonic mean between recall and precision. The higher the value of these measures, the better the clustering is for WSD.

Figure 6 shows the values of recall, precision and F1 achieved over all sentences by varying β_0 threshold. Each β_0 corresponds to a percentile value of the pair-wise similarities of the senses. As it can be appreciated, the extended star clustering algorithm produces stable results up to 70th percentile. The very high recall values obtained in this experiment (around 0.98) demonstrate the usefulness of the sense groups produced by the extended star algorithm.

The relatively low values of precision (around 0.38) were expected because no refinement of the clustering was performed in the experiment (i.e., just one iteration was done).

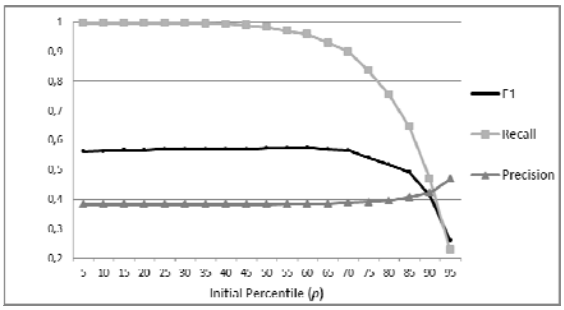

Figure 6: Performance of the extended star clustering in the identification of semantically related groups of senses for WSD

5.2. Sense Clustering and Clustering Refinement for WSD

The goal of the second experiment is to explore the role of both: clustering in the disambiguation process, and the iterative process of clustering refinement in the disambiguation.

Thus, we compare our clustering-based approach with a non-clustering based WSD algorithm obtained as an instance of the disambiguation framework by using the trivial clustering algorithm (i.e., the set of singletons consisting of a word sense) and the same filter and stopping criterion of our proposal. Note that the non-clustering approach selects the senses having maximum similarity w.r.t. the context instead of sense groups. This approach resembles those strategies based on the gloss overlap and relatedness-based measures.

For evaluating the impact of the iterative clustering refinement strategy we consider the results of our approach just after each iteration (i.e., by regarding the remaining senses as the right ones). Figure 7 summarizes the results of this experiment using different p values in the disambiguation process.

The clustering-based method outperforms the non-clustering one for all percentile values. This confirms our hypothesis that clustering provides a way to identify groups of semantically related word senses that can be useful for disambiguation tasks. Nevertheless, it is important to notice the relatively high

impact of the sense representation on the non-clustering baseline.

We can also observe that starting from an initial clustering of senses, the disambiguation results are clearly improved after performing each refinement iteration. This corroborates the idea of using the iterative process of clustering refinement.

Another interesting observation is that the accuracy of our approach is fairly consistent for all percentile values (the F1 scores remain between 0.702 and 0.722). The best F1 score was 0.722 for a 65th percentile. Thus, the greater the percentile value, a lower number of iterations is required to satisfy the stopping criterion.

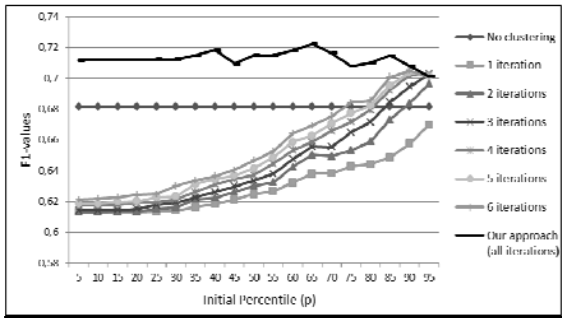

Figure 7: Performance of the star-based WSD algorithm vs. the non-clustering based method

5.3. Extended Star Clustering vs. WordNet Domains

As previously mentioned, WordNet domains have been widely used by several disambiguation algorithms. As WordNet domains induce a clustering distribution for word senses, the purpose of this experiment is to evaluate its performance in the clustering-based WSD framework. With this aim, we replace the clustering component of the star-based WSD algorithm with the clustering induced by WordNet domains.

The induced clustering considers each domain different from *Factotum* as a cluster, that is, all word senses labeled with a domain *d* (*d* ≠ *Factotum*) in a sentence belong to the same cluster. Also, all the senses of a word *w* labeled with *Factotum* domain are considered as belonging to all clusters that do not cover *w*.

In order to define an appropriate clustering refinement strategy, we consider the different levels of the domain hierarchy to generate the word sense clusters. Thus, three iterations are carried out. The first one only considers the global domains of the hierarchy. The second one relies on the basic domains, and finally the domain hierarchy leaves (i.e., the most specific domains) are regarded.

Table 2 shows the results of this experiment. It is shown that, the extended star clustering performs better than the method based on WordNet domains. It is worth mentioning that the results obtained by WordNet domains-based method also support the idea of using the clustering refinement strategy for improving the disambiguation precision.

Method	F1-value
Global domains	0.624
Basic domains	0.631
Domain hierarchy leaves	0.632
Star-based approach	**0.722**

Table 2: Performance of the star- vs. WordNet domains-based WSD algorithms

5.4. Evaluation on standard data sets

In order to contextualize our approach in the current disambiguation state-of-the-art, we evaluate our proposal on several benchmark data sets. Specifically, we use SemCor (Miller et al., 1993), Senseval-3 (Snyder & Palmer, 2004) and SemEval-2007 corpora (see details in http://www.senseval.org/). Our experiments were run in the all-words setting, where the algorithm must disambiguate all (content) words in a given document.

In this paper, we use a subset of SemCor 2.0 composed by all the documents of brown1 and brown2 corpora. It contains a total of 192,639 words (88,058 nouns, 48,328 verbs, 35,664 adjectives and 20,589 adverbs) tagged with WordNet 2.0 senses. In the case of Senseval-3, we use the all-words corpus composed by 2081 words (951 nouns, 751 verbs, 364 adjectives and 15 adverbs) annotated with WordNet 2.0.

In particular, we use the SemEval-2007 Task 7 (Navigli et al., 2007), and Task 17 (Pradhan et al., 2004) data sets. The Task 7 data set consists of 5,377 words of five articles (the first three in common with Task 17) obtained from the WSJ corpus, Wikipedia and Amy Steedman's Knights of the Art. 2,269 of these words are annotated with WordNet 2.1 senses in Task 7 and 455 (159 nouns and 296 verbs) in Task 17.

A fine-grained disambiguation was evaluated in SemCor, Senseval-3 and SemEval 2007 Task 17 corpora, whereas a coarse-

grained evaluation was done in the corpus provided by Task 7 of SemEval-2007. Our results are summarized in Table 3.

In SemEval-2007 competition we participated with the system TKB-UO (Anaya-Sánchez, Pons-Porrata, & Berlanga-Llavori, 2007), which is a previous version of the proposed method in this paper. This system was considered the best unsupervised system (Navigli, 2009). The poor performance of our proposal in the task 17 can be explained by the high polysemy degree of verbs and its relatively small number of relations in WordNet.

Recently, Navigli and Lapata (2010) evaluated several graph-based WSD methods and well- known baselines on these benchmark data sets. From these results, we find that our proposal yields competitive performance with the state-of-the-art unsupervised WSD methods.

Data set	F1-value
SemEval-2007 Task 7 (TKB-UO)	0.702
SemEval-2007 Task 17 (TKB-UO)	0.325
SemEval-2007 Task 7	**0.722**
SemEval-2007 Task 17	**0.332**
Senseval-3 all-words	0.428
SemCor	0.498

Table 3: Our results on standard data sets

6. Conclusions

In this paper, the star-based disambiguation algorithm has been introduced. This unsupervised and knowledge-based method is derived from the framework proposed in (Anaya-Sánchez et al., 2006) by using both feature vectors built from WordNet for representing word senses, and the extended star clustering algorithm. Unlike previous work in WSD, the clustering algorithm is here used to contextually group word senses according to their representations. Also, our approach profits from both sense vectors and clustering method to overcome the sparseness of WordNet relations for associating semantically related word senses.

As a result, our proposal not only is able to disambiguate all words in a sentence but also reveals the implicit relationships (not asserted in WordNet) existing among these word senses. These relations can provide evidence for the sense choices and strong clues that can be helpful for manual annotators (Navigli & Velardi, 2005).

The experiments carried out over the coarse-grained English all-words corpus of SemEval-2007 validate both the use of the extended star clustering for connecting semantically related word senses, and the iterative clustering refinement strategy for WSD. We have also shown that the proposed scheme for grouping word senses outperforms those induced by WordNet domains at any of its abstraction levels. Despite our method requires a percentile value as input parameter, we demonstrate that its accuracy is fairly consistent for almost percentile values.

Our proposal performs the best among all the unsupervised systems participating in the Task 7 of SemEval-2007 competition. It also achieves competitive results with respect to the state-of-the-art unsupervised WSD methods on existing benchmark data sets.

One of the most critical issues for clustering word senses is the representation of senses. As future work, we plan to enrich word sense vectors with other external resources (e.g., Wikipedia), in order to evaluate if they produce better disambiguation results. In particular, a proper sense representation for verbs is a key issue we must face to. Future work also regards to explore the role of the filtering component in the disambiguation and to enrich the information about the textual context. Finally, we will examine the impact of sense co-occurrences that can be obtained from lexical resources, like extended WordNet, in the clustering process.

7. References

Agirre, E., & López, O. (2003). Clustering WordNet Word Senses. In *Proceedings of the Conference on Recent Advances on Natural Language Processing (RANLP '03)*, pp. 121-130. Borovets, Bulgaria.

Anaya-Sánchez, H., Pons-Porrata, A., & Berlanga-Llavori, R. (2006). Word Sense Disambiguation based on Word Sense Clustering. In Simão Sichman, J., Coelho, H., & Oliveira Rezende, S. (Eds.), *Proceedings of 10th Ibero-American Conference on Artificial Intelligence (IBERAMIA-SBIA'2006)*, pp. 472-481. Berlin, Germany: Springer.

Anaya-Sánchez, H., Pons-Porrata, A., & Berlanga-Llavori, R. (2007). TKB-UO:

Using Sense Clustering for WSD. In *Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval'2007)*, pp. 322-325. Morristown, NJ: ACL.

Aslam, J., Pelekhov, E., & Rus, D. (2004). The Star Clustering Algorithm for Static and Dynamic Information Organization. *Journal of Graph Algorithms and Applications, 8*, 95-129.

Brody, S., Navigli, R., & Lapata, M. (2006). Ensemble Methods for Unsupervised WSD. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (COLING-ACL'2006)*, pp. 97-104. Morristown, NJ: Association for Computational Linguistics.

Gil-García, R., Badía-Contelles, J.M., & Pons-Porrata, A. (2003). Extended Star Clustering Algorithm. In Sanfeliu, A., & Ruiz-Shulcloper, J. (Eds.), *Proceedings of the 8th Iberoamerican Congress on Pattern Recognition (CIARP'2003)*, pp. 480-487. Berlin, Germany: Springer.

Kolte, S.G., & Bhirud, S.G. (2008). Word Sense Disambiguation Using WordNet Domains. In *Proceedings of the First International Conference on Emerging Trends in Engineering and Technology (ICETET'08)*, pp. 1187-1191. Washington, DC: IEEE Computer Society.

Magnini, B., & Cavaglià, G. (2000). Integrating Subject Field Codes into WordNet. In Gavrilidou, M., Crayannis, G., Markantonatu,S.,Piperidis,S.,&Stainhaouer, G. (Eds.), *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC'2000)*, pp. 1413-1418. Athens, Greece.

Magnini, B., Strapparava, C., Pezzulo, G., & Gliozzo, A. (2002). The role of domain information in Word Sense Disambiguation. *Natural Language Engineering, 8(4)*, 359-373.

Mihalcea, R. (2005). Unsupervised Large-Vocabulary Word Sense Disambiguation with Graph-Based Algorithms for Sequence Data Labeling. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing (HLT/EMNLP'2005)*, pp. 411-418. Morristown, NJ: Association for Computational Linguistics.

Miller, G. (1995). WordNet: A Lexical Database for English. *Communications of the ACM, 38(11)*, 39-41.

Miller, G., Leacock, C., Randee, T., & Bunker, R. (1993). A Semantic Concordance. In *Proceedings of the Workshop on Human Language Technology (HLT'93)*, pp. 303-308. Morristown, NJ: Association for Computational Linguistics.

Navigli, R. (2006). Meaningful Clustering of Senses Helps Boost Word Sense Disambiguation Performance. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (COLING-ACL'2006)*, pp. 105-112. Morristown, NJ: Association for Computational Linguistics.

Navigli, R. (2009). Word Sense Disambiguation: a Survey. *ACM Computing Surveys, 41(2)*, 1-69.

Navigli, R., & Lapata, M. (2010). An Experimental Study of Graph Connectivity for Unsupervised Word Sense Disambiguation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 32(4)*, 678-692.

Navigli, R., Litkowski, K.-C., & Hargraves, O. (2007). SemEval-2007 Task 07: Coarse-Grained English All-Words Task. In *Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval'2007)*, pp. 30-35. Morristown, NJ: Association for Computational Linguistics.

Pedersen, T. (2006). Unsupervised Corpus Based Methods for WSD. In Agirre E., & Edmonds, P. (Eds.), *Word Sense Disambiguation: Algorithms and Applications, chapter 6*. Springer.

Pradhan, S.S., Loper, E., Dligach, D., & Palmer, M. (2007). SemEval-2007 Task 17: English Lexical Sample, SRL and All Words. In *Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval'07)*, pp. 87-92. Morristown, NJ: Association for Computational Linguistics.

Snyder, B., & Palmer, M. (2004). The English All-Words Task. In *Proceedings of Senseval-3: The Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pp. 41-43. New Brunswick, NJ: Association for Computational Linguistics.