An Enhanced Lesk Word Sense Disambiguation Algorithm through a Distributional Semantic Model

Pierpaolo Basile **Annalina Caputo** **Giovanni Semeraro**
Department of Computer Science, University of Bari Aldo Moro
Via E. Orabona, 4, Bari - 70125 Italy
{pierpaolo.basile,annalina.caputo,giovanni.semeraro}@uniba.it

Abstract

This paper describes a new Word Sense Disambiguation (WSD) algorithm which extends two well-known variations of the Lesk WSD method. Given a word and its context, Lesk algorithm exploits the idea of maximum number of shared words (maximum overlaps) between the context of a word and each definition of its senses (gloss) in order to select the proper meaning. The main contribution of our approach relies on the use of a word similarity function defined on a distributional semantic space to compute the gloss-context overlap. As sense inventory we adopt BabelNet, a large multilingual semantic network built exploiting both WordNet and Wikipedia. Besides linguistic knowledge, BabelNet also represents encyclopedic concepts coming from Wikipedia. The evaluation performed on SemEval-2013 Multilingual Word Sense Disambiguation shows that our algorithm goes beyond the most frequent sense baseline and the simplified version of the Lesk algorithm. Moreover, when compared with the other participants in SemEval-2013 task, our approach is able to outperform the best system for English.

1 Introduction

Unsupervised Word Sense Disambiguation (WSD) algorithms aim at resolving word ambiguity without the use of annotated corpora. Among these, two categories of knowledge-based algorithms gained popularity: overlap- and graph-based methods. The former owes its success to the simple intuition underlying that family of algorithms, while the diffusion of the latter started growing after the development of semantic networks.

The overlap-based algorithms stem from the Lesk (1986) one, which inspired a whole family of methods that exploit the number of common words in two sense definitions (*glosses*) to select the proper meaning in a context. Glosses play a key role in Lesk algorithm, which exploits only two types of information: 1) the set of dictionary entries, one for each possible word meaning, and 2) the information about the context in which the word occurs. The idea is simple: given two words, the algorithm selects those senses whose definitions have the maximum overlap, i.e. the highest number of common words in the definition of the senses. In order to extract definitions, Lesk adopted the *Oxford Advanced Learner's* dictionary. This approach suffers from two problems: 1) complexity, the number of comparisons increases combinatorially with the number of words in a text; and 2) definition expressiveness, the overlap is based only on word co-occurrences in glosses. The first problem was tackled by a "simplified" version of Lesk algorithm (Kilgarriff and Rosenzweig, 2000), which disambiguated each word separately. Given a word, the meaning whose gloss shows the maximum overlap with the current context, represented by the surrounding words, is selected. The simplified Lesk significantly outperforms the original Lesk algorithm as proved by Vasilescu et al. (2004). The second problem was faced by Banerjee and Pedersen (2002), who proposed an "adapted" Lesk algorithm. The adapted variation exploits relationships among meanings: each gloss is extended by the definitions of semantically related meanings. Banerjee and Pedersen adopt WordNet as semantic network and their algorithm takes into account several relations:

Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, pages 1591–1600, Dublin, Ireland, August 23-29 2014.

hypernym, hyponym, holonym, meronym, troponym and attribute relation. The adapted algorithm outperforms the Lesk one in disambiguating nouns in SensEval-2 English task. Despite these improvements, overlap-based algorithms failed to stand the pace with figures achieved by graph approaches. Their ability to disambiguate all words in a sequence at once, meanwhile exploiting the existing interconnections (edges) between senses (nodes), has made these algorithms more principled than methods that use the sense definition overlaps. Moreover, the success of PageRank in web search has inspired a new vein of algorithms for sense disambiguation that blossomed during the past years. Although graph-based algorithms have taken advantage of the rich set of relationships available in dictionaries like WordNet, they completely neglected the role of glosses in the disambiguation process.

From our standpoint, glosses are an important piece of information since they extensionally define a word meaning. In this paper we propose a revised version of the simplified and adapted Lesk variations that overcomes limits due to the definition expressiveness. Our method replaces the concept of overlap with that of similarity. Similarity is computed on a Distributional Semantic Space (DSS) in order to account for semantic relationships between words occurring in the definition and context, for as they emerge from the use in the language. Indeed, Distributional Semantics Models (DSM) exploit the geometric metaphor of meanings, which are represented through points into a space where distance is a measure of semantic similarity. The point representation inherits information about all co-occurring context words, and then it is suitable for computing the overlap where no exact word matching can occur. In addition, we introduce two functions: the former assigns an inverse gloss frequency (IGF) score to each term occurring in the extended gloss, the latter exploits information about sense frequencies extracted from an annotated corpus.

We choose BabelNet (Navigli and Ponzetto, 2012) as sense inventory. BabelNet is a very large semantic network built up exploiting both WordNet and Wikipedia. Besides linguistic knowledge, it also represents encyclopedic concepts coming from Wikipedia and information about named entities. This makes our approach inherently multilingual and suitable for tasks such as named entity disambiguation. The evaluation on SemEval-2013 Multilingual Word Sense Disambiguation (Navigli et al., 2013) proves that our method is able to outperform both baselines (simplified Lesk and most frequent sense) and, for English language, also the best SemEval-2013 participant.

The paper is structured as follows. Section 2 provides a brief introduction to Distributional Semantic Models, while Section 3 describes the proposed methodology. Evaluation and details about the algorithm implementation are reported in Section 4, while related work is described in Section 5. Finally, conclusions close the paper.

2 Distributional Semantic Models

Semantic (or *Word*) *Spaces* are geometrical spaces of words where vectors express concepts, and their proximity is a measure of the semantic relatedness. One of their greatest virtues is that they can be built using entirely unsupervised analysis of free text. Moreover, they make few language-specific assumptions since only tokenized text is needed. *WordSpace*s are inspired by the *distributional hypothesis* (Harris, 1968) whereby the meaning of a word is determined by the rules of its use in the context of ordinary and concrete language behaviour. This means that words are semantically similar if they share *contexts* (surrounding words). Building a *WordSpace* involves the definition of a distributional model, that is a quadruple (Lowe, 2001) consisting of: the space basis (word vectors) and dimension; the function that takes into account word co-occurrences and how these are represented in the final vector; a similarity function defined over vectors; and eventually a map that transforms the vector space.

Our idea is to apply DSMs to WSD for computing the overlap between the gloss of the meaning and the context as a similarity measure between their corresponding vector representations in a *SemanticSpace*. In this paper we build a *SemanticSpace* (co-occurrences matrix M) by analysing the distribution of words in a large corpus, then M is reduced using Latent Semantic Analysis (LSA) (Landauer and Dumais, 1997). LSA collects the text data in a co-occurrence matrix, which is then decomposed into smaller matrices with Singular-Value Decomposition (SVD). Hence, LSA represents high-dimensional vectors in a lower-dimensional space while capturing latent semantic structures in the text data.

Given the vector representation of two words, DSMs usually compute their similarity as the cosine of the angle between them. In our case, the gloss and the context are composed by several terms, so in order to compute their similarity we need a method to compose the words occurring in these sentences. It is possible to combine words through vector addition ($+$) that corresponds to the point-wise sum of vector components. For each set of terms, phrase or sentence, we build its vector representation by adding the vectors associated to the words it is composed of. Then, the similarity measure is computed as the cosine similarity between the two phrases/sentences. More formally, if $g = g_1g_2...g_n$ and $c = c_1c_2...c_m$ are the gloss and the context respectively, we build their vector representation $\mathbf{g}$ and $\mathbf{c}$ in the *SemanticSpace* through addition of word vectors belonging to them:

$$\begin{aligned} \mathbf{g} &= \mathbf{g_1} + \mathbf{g_2} + \ldots + \mathbf{g_n} \\ \mathbf{c} &= \mathbf{c_1} + \mathbf{c_2} \ldots + \mathbf{c_m} \end{aligned} \tag{1}$$

The cosine similarity between $\mathbf{g}$ and $\mathbf{c}$ is a measure of the similarity of the two sentences that we consider as a score associated to the candidate meaning with respect to the context.

3 Methodology

At the heart of our approach there is the simplified Lesk algorithm. Given a text $w_1w_2...w_n$ of n words, we disambiguate one at a time taking into account the similarity between the gloss associated to each sense of the target word w_i and the context. The meaning whose gloss has the highest similarity is selected. The context could be represented by a subset of surrounding words or the whole text where the word occurs. Moreover, taking into account the idea of the Banerjee's adaptation, we expand each gloss with those of related meanings.

Our sense inventory is BabelNet, a very large multilingual semantic network built relying on both WordNet and Wikipedia. In BabelNet linguistic knowledge is enriched with encyclopedic concepts coming from Wikipedia. WordNet synsets and Wikipedia concepts (pages) are connected in an automatic way. We choose BabelNet for three reasons: 1) glosses are richer and contain text from Wikipedia, 2) it is multilingual, thus the proposed algorithm can be applied to several languages, and 3) it also contains information about named entities, thus an algorithm using BabelNet could be potentially used to disambiguate entities.

Our algorithm consists of five steps:

1. **Look-up**. For each word w_i, the set of possible word meanings is retrieved from BabelNet. First, we look for senses coming from WordNet (or WordNet translated into languages different from English). If no sense is found, we retrieve senses from Wikipedia. We adopt this strategy because mixing up all senses from Wikipedia and WordNet results in worse performance. Conversely, if a word does not occur in WordNet it is probably a named entity, thus Wikipedia could provide useful information to disambiguate it.

2. **Building the context**. The context C is represented by the l words to the left and to the right of w_i. We also adopt a particular configuration in which the context is represented by all the words that occur in the text.

3. **Gloss expansion**. We indicate with s_{ij} the j-th sense associated to the target word w_i. We expand the gloss g_{ij} that describes the j-th sense using the function "getRelatedMap" provided by BabelNet API. This method returns all the meanings related to a particular sense. For each related meaning, we retrieve its gloss and concatenate it to the original gloss g_{ij} of s_{ij}. During this step we remove glosses belonging to synsets related by the "antonym" relationship. The result of this step is an extended gloss denoted by g^*_{ij}. In order to give more importance to terms occurring in the original gloss, the words in the expanded gloss are weighed taking into account both the distance between s_{ij} and the related synsets and the word frequencies. More details about term scoring are reported in Subsection 3.2.

4. **Building semantic vectors**. Exploiting the DSM described in Section 2, we build the vector representation for each gloss g^*_{ij} associated with the senses of w_i and the context C.

5. **Selecting the correct meaning**. For each gloss g^*_{ij}, the algorithm computes the cosine similarity between its vector representation and context vector C. The similarity is linearly combined with the probability $p(s_{ij}|w_i)$ that takes into account the sense distribution of s_{ij} given the word w_i; details are reported in Subsection 3.1. The sense with the highest similarity is chosen.

In order to compare our approach to the simplified Lesk algorithm, we developed a variation of our method in which, rather than building the semantic vectors, we count the common words between each extended gloss g^*_{ij} and the context C. In this case, we apply stemming to maximize the overlap.

3.1 Sense Distribution

The selection of the correct meaning takes also into account the senses distribution of the word w_i. We retrieve information about sense occurrences from WordNet (Fellbaum, 1998), which reports for each word w_i its sense inventory S_i with the number of times that the word w_i was tagged with s_{ij} in SemCor. SemCor is a collection of 352 documents manually annotated with WordNet synsets. We introduce the sense distribution factor in order to consider the probability that a word w_i can be tagged with the sense s_{ij}. Moreover, since some synsets do not occur in SemCor and can cause zero probabilities, we adopt an additive smoothing (also called Laplace smoothing). Finally the probability is computed as follow:

$$p(s_{ij}|w_i) = \frac{t(w_i, s_{ij}) + 1}{\#w_i + |S_i|} \tag{2}$$

where $t(w_i, s_{ij})$ is the number of times the word w_i is tagged with s_{ij} and $\#w_i$ is the number of occurrences of w_i in SemCor.

3.2 Gloss Term Scoring

The extended gloss conflates words from the gloss directly associated with the synset s_{ij} with those of the glosses appearing in the related synsets. When we add words to the extended gloss, we weigh them by a factor inversely proportional to the distance in the graph (number of edges) between s_{ij} and the related synsets so to reflect their different origin. Let d be that distance, then the weight is computed as $\frac{1}{1+d}$. Finally, we re-weigh words using a strategy similar to the inverse document frequency (IDF) that we call inverse gloss frequency (IGF). The idea is that if a word occurs in all the extended glosses associated with a word, then it poorly characterizes the meaning description. Let gf^*_k be the number of extended glosses that contain a word w_k, then IGF is computed as follow:

$$IGF_k = 1 + log_2 \frac{|S_i|}{gf^*_k} \tag{3}$$

This approach is similar to the idea proposed by Vasilescu et al. (2004), where TF-IDF of terms is computed taking into account the glosses in the whole WordNet, while we compute IGF considering only the glosses associated to each word. Finally, the weight for the word w_k appearing h times in the extended gloss g^*_{ij} is given by:

$$weight(w_k, g^*_{ij}) = h \times IGF_k \times \frac{1}{1+d} \tag{4}$$

4 Evaluation

The evaluation is performed using the dataset provided by the organizers of the Task-12 of SemEval-2013. The task concerns Multilingual Word Sense Disambiguation, a traditional WSD “all-words” experiment in which systems are expected to assign the correct BabelNet synset to all occurrences of noun phrases (which refer to both disambiguated nouns and named entities) within arbitrary texts in different languages.

The goal of our evaluation is twofold: to prove that our strategy outperforms the simplified Lesk approach, and then to compare our system with respect to the other task participants.

In both the experiments we consider two languages, English and Italian, to evaluate performance in a multilingual setting. It is important to underline that in our approach only stemming and the corpus used to build the distributional model are language dependent.

4.1 System Setup

Our method[1] is completely developed in JAVA using BabelNet API 1.1.1 provided by the authors[2]. We adopt the standard Lucene analyzer to tokenize both glosses and the context, while Snowball library[3] is used for stemming in several European languages. The *SemanticSpace*s for the two languages are built using proprietary code relying on two Lucene indexes, which contain documents from British National Corpus (BNC) for English, and from Wikipedia dump for Italian, respectively. For each language, the co-occurrences matrix M contains information about the top 100,000 most frequent words in the corpus. M is reduced by LSA using the SVDLIBC tool[4] and setting a reduced dimension equal to 200. The result of the SVD decomposition is stored in a binary format used by our algorithm implementation.

It is important to underline that BabelNet Italian glosses are taken from MultiWordNet, which does not contain glosses for all the synsets. Then, we replace each missing gloss by the words that belong to the synset.

We evaluate our system by setting two parameters: 1) the context size (3, 5, 10, 20 and the whole text); 2) the use of information about sense distribution (see Formula (2) in Subsection 3.1).

The gloss term scoring function is always applied, since it provides better results. The synset distance d used to expand the gloss is fixed to 1; we experimented with a distance d set to 2 without any improvement. The sense distribution is linearly combined with the cosine similarity score through a coefficient set to 0.5.

Some notes about sense frequency: by using only sense distribution to select a sense we obtain an algorithm that performs like the most frequent sense (MFS). In other words, the algorithm always assigns the most probable meaning. It is well known that this approach obtains very good performance and it is hard to be outperformed especially by unsupervised approaches.

4.2 English Evaluation

Table 1 reports results of our algorithm (DSM) compared with the best simplified Lesk approaches (LESK) in terms of precision (P), recall (R), F-measure (F) and attempt (A). Attempt is the percentage of words disambiguated by the algorithm. $SenseDistr.$ column reports the information about when the sense distribution formula (see Subsection 3.1) is used (Y) or not (N); it is also important to point out that MSF produces the same results of using only sense distribution i.e. the first sense is the most likely one. We have experimented different context sizes also for the Lesk algorithm, although for the sake of readability we report only the best Lesk with and without sense distribution.

All our runs always obtain an attempt of 100%; thus the precision and recall values are always the same. The run **EN.DSM.10** obtains the best result using both sense distribution information and the whole text (W) as context. Another important outcome is the result obtained by the run **EN.DSM.5** that, without information about sense distribution, is able to overcome the MFS baseline. To the best of our knowledge, this is the first completely unsupervised system able to overcome the MFS baseline without using sense frequency. Both results (**EN.DSM.10** and **EN.DSM.5**) suggest that the vector representation of the whole text helps the system to achieve the best performance.

Considering the Lesk method, generally, the best size for the context is 3, then a larger set of surrounding words results in worse performance, differently to what happens in the distributional approach. This is probably due to the fact that words distant from the target one match some incorrect glosses. It is important to note that no simplified Lesk run is able to overcome the MFS baseline.

[1] Available on line: `https://github.com/pippokill/lesk-wsd-dsm`

[2] Available on line: `http://lcl.uniroma1.it/babelnet/download.jsp`

[3] Available on line: `http://snowball.tartarus.org/`

[4] Available on line: `http://tedlab.mit.edu/\textasciitildedr/SVDLIBC/`

Run	$ContextSize$	$SenseDistr.$	P	R	F	A
MFS	-	-	0.656	0.656	0.656	100%
EN.LESK.1	3	N	0.525	0.525	0.525	100%
EN.LESK.6	3	Y	0.633	0.633	0.633	100%
EN.DSM.1	3	N	0.536	0.536	0.536	100%
EN.DSM.2	5	N	0.605	0.605	0.605	100%
EN.DSM.3	10	N	0.633	0.633	0.633	100%
EN.DSM.4	20	N	0.650	0.650	0.650	100%
EN.DSM.5	W	N	0.687	0.687	**0.687**	100%
EN.DSM.6	3	Y	0.669	0.669	0.669	100%
EN.DSM.7	5	Y	0.677	0.677	0.677	100%
EN.DSM.8	10	Y	0.689	0.689	0.689	100%
EN.DSM.9	20	Y	0.696	0.696	0.696	100%
EN.DSM.10	W	Y	0.715	0.715	**0.715**	100%

Table 1: Results of the English evaluation.

Comparing DSM-based with simplified Lesk approaches, the former consistently outperform Lesk-based algorithms when considering the use (or not) of sense distribution.

4.3 Italian Evaluation

Table 2 reports results of our algorithm for the Italian language. In this case we still obtain the best result (**IT.DSM.10**) using DSM and sense distribution. As for English, the systems without sense distribution overcome the MFS baseline. However, in this case simplified Lesk with sense distribution is able to outperform the MFS. We ascribe this different behaviour to the problem of missing glosses that we solved by adding the words in the synset.

Run	$ContextSize$	$SenseDistr.$	P	R	F	A
MFS	-	-	0.572	0.572	0.572	100%
IT.LESK.2	5	N	0.531	0.530	0.530	99.71%
IT.LESK.10	W	Y	0.608	0.606	0.607	99.71%
IT.DSM.1	3	N	0.611	0.609	0.610	99.71%
IT.DSM.2	5	N	0.608	0.607	0.607	99.71%
IT.DSM.3	10	N	0.627	0.625	0.626	99.71%
IT.DSM.4	20	N	0.629	0.627	0.628	99.71%
IT.DSM.5	W	N	0.634	0.632	**0.633**	99.71%
IT.DSM.6	3	Y	0.632	0.630	0.631	99.71%
IT.DSM.7	5	Y	0.631	0.629	0.630	99.71%
IT.DSM.8	10	Y	0.636	0.634	0.635	99.71%
IT.DSM.9	20	Y	0.640	0.638	0.639	99.71%
IT.DSM.10	W	Y	0.642	0.640	**0.641**	99.71%

Table 2: Results of the Italian evaluation.

4.4 Task Results

In this subsection, we report our best performance (Table 3) with respect to the other participants in the SemEval-2013 Task-12 on multilingual Word Sense Disambiguation, for both English (Table 3a) and Italian (Table 3b).

Regarding the English language, our best methods are able to outperform all the systems. It is important to underline that our method without knowledge about sense distribution (**EN.DSM.5**) outperforms both the MFS and all the task participants. This is a very important outcome because generally

System	*F*
EN.DSM.10	**0.715**
EN.DSM.5	**0.687**
UMCC-DLSI-2	0.685
UMCC-DLSI-3	0.680
UMCC-DLSI-1	0.677
MFS	*0.656*
DAEBAK	0.604
GETALP-BN-1	0.263
GETALP-BN-2	0.266

(a) English

System	*F*
UMCC-DLSI-2	0.658
UMCC-DLSI-1	0.657
IT.DSM.10	**0.641**
IT.DSM.5	**0.633**
DAEBAK	0.613
MFS	*0.572*
GETALP-BN-2	0.325
GETALP-BN-1	0.324

(b) Italian

Table 3: Results of our best systems with respect to the Semeval-2013 participants.

knowledge-base approaches without information about sense frequencies obtain low results. For example, the UMCC-DLSI system (Gutiérrez et al., 2013) exploits sense frequency to modify prior probability of synset nodes in the PageRank, and DAEBAK system (Manion and Sainudiin, 2013) uses MFS when it is not able to select a meaning. Our experiments show that a dictionary-based approach and the adoption of a distributional semantic model for computing the similarity are enough to obtain good results. Moreover, by adding information about sense frequencies we are able to boost our performance and obtain over 70% of F-measure.

For Italian, our systems are not able to reach the same performance as for English, although they still outperform the MFS and two task participants. We think that these results are due to the same problem observed for the Italian evaluation (Subsection 4.3), that is to say, the poor quality of glosses.

5 Related Work

WSD has been an active area of NLP whose roots stem from early work in Machine Translation. Ambiguity resolution has been pursued as a way to improve retrieval systems, and generally to get better information access. Despite its ancient roots and perceived importance, this task is still far from being resolved.

Our WSD method relies on both the Lesk algorithm and its two variants: simplified (Kilgarriff and Rosenzweig, 2000) and adapted (Banerjee and Pedersen, 2002). Several approaches have modified the Lesk algorithm to reduce is exponential complexity, like the one based on Simulated Annealing (Cowie et al., 1992). Basile et al. (2007) adopted the simplified Lesk algorithm to disambiguate adjectives and adverbs, combining it with other two methods for nouns and verbs: the combination of different approaches for each part-of-speech resulted in better performance with respect to the use of a single strategy. More recently, Schwab et al. (2013) proposed GETALP, another unsupervised WSD algorithm inspired by Lesk. Their approach computes a local similarity using the classical Lesk measure (overlap between glosses), and then the local similarity is propagated to the whole text (global similarity) using an algorithm inspired by the Ant Colony. This approach got the lowest results during the SemEval-2013 Task 12 evaluation due to a bug in the system. However, the correct implementation achieves 0.583 of F-measure for English and 0.528 for Italian.

Another problem with the Lesk-based approaches is to maximize the chances of overlap between glosses or between the gloss and the context. To solve this problem, Ponzetto and Navigli (2010) extended WordNet with Wikipedia pages (WordNet++) in order to produce a richer lexical resource, obtaining the English portion of BabelNet. The simple Lesk algorithm built over WordNet++ outperformed the WordNet-base version, although it was not successful in overtaking the MFS baseline. Our approach tries to extend glosses using related synsets and adopts distributional semantics to address the problem of data sparsity. A different perspective has been recently proposed by Wang and Hirst (2014), where the limits of the exact string matching between glosses and context are overcome by a Naive Bayes-based similarity measure.

Other unsupervised approaches rely on graph algorithms that exploit the graph generated by a semantic network where the senses are connected through semantic relations. For example, DAEBAK (Manion and Sainudiin, 2013) adopts a sub-graph of BabelNet generated taking into account the surrounding words of the target word. A measure of connectivity computed on the sub-graph is used to extract the most probable sense. The MFS is used when the algorithm is not able to choose any sense. Also Navigli and Lapata (2010) exploit the idea of graph connectivity measures for identifying the most important node (sense) in the graph. Experiments conducted on SemCor show that the Degree Centrality provides best results compared to other well known techniques, such as PageRank, HITS, Key Player Problem and Betweenness Centrality. Graph-based methods also showed their validity during the SemEval 2013 Multilingual Word Sense Disambiguation task. The best system, UMCC-DLSI (Gutiérrez et al., 2013), builds a graph using several resources: WordNet, WordNet Domains and the eXtended WordNet. Then, the best sense is selected using the PageRank algorithm where the a priori probability of senses is estimated according to the sense frequencies. This is an extension of UBK algorithm (Agirre and Soroa, 2009), the first application of personalized PageRank to the WSD problem.

On the distributional side, Brody and Lapata (2008) use distributional similarity to automatically annotate a corpus for training a supervised method. Each target word in the corpus is paired with a list of neighbours selected via distributional similarity. A neighbour is linked to a sense in WordNet and then it is used for the annotation. Differently from our approach, distributional similarity is used to automatically annotate a training corpus rather than directly disambiguate terms. Miller et al. (2012) exploit a distributional thesaurus to expand both glosses and the context, then they apply the classical word overlap adopted in the simplified Lesk. This approach is strongly related to our, although our approach directly computes the overlap in the geometric space that implements the distributional semantic model. In particular, we build a vector representation for both the gloss and the context. In recent years, other approaches have tried to solve unsupervised WSD relying on distributional information. Gliozzo et al. (2005) build a matrix taking into account words and WordNet domains. The matrix is reduced using LSA and then it is combined in a kernel exploited in a supervised approach. Martinez et al. (2008) propose a method based on topic signatures automatically constructed using the Web, while Li et al. (2010) adopt Latent Dirichlet Allocation (LDA) to compute a conditional probability between a sense and the context. In this model, a sense is represented by its paraphrases used to build the LDA model.

6 Conclusions and Future Work

In this paper we describe an unsupervised WSD approach which selects the best sense according to the distributional similarity with respect to the context. In particular, the similarity is computed representing both the gloss and the context as vectors in a geometric space generated by a distributional semantic model based on LSA. The evaluation, conducted on the Task-12 of SemEval-2013, shows promising results: our method is able to overcome both the most frequent sense baseline and, for English, also the other task participants. We provide two implementations of our approach, with and without exploiting sense frequencies. For English, both implementations outperform the SemEval-2013 participants and the MFS. Differently, for Italian both implementations do not reach the figures of the best participant, but they are able to defeat the MFS. As future work, we plan to extend our evaluation to other languages, and to investigate how to adapt our approach to a specific domain. In particular, distributional models built upon a domain corpus, and sense frequencies extracted from the same corpus, could result in a domain adaptation of our algorithm.

Acknowledgements

This work fulfils the research objectives of the projects PON 01_00850 ASK-Health (Advanced System for the interpretation and sharing of knowledge in health care) and PON 02_00563_3470993 project “VINCENTE - A Virtual collective INtelligenCe ENvironment to develop sustainable Technology Entrepreneurship ecosystems” funded by the Italian Ministry of University and Research (MIUR).

References

Eneko Agirre and Aitor Soroa. 2009. Personalizing PageRank for Word Sense Disambiguation. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 33–41, Athens, Greece, March. Association for Computational Linguistics.

Satanjeev Banerjee and Ted Pedersen. 2002. An Adapted Lesk Algorithm for Word Sense Disambiguation Using WordNet. In Alexander Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing*, volume 2276 of *Lecture Notes in Computer Science*, pages 136–145. Springer Berlin Heidelberg.

Pierpaolo Basile, Marco de Gemmis, Anna Lisa Gentile, Pasquale Lops, and Giovanni Semeraro. 2007. UNIBA: JIGSAW algorithm for Word Sense Disambiguation. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 398–401, Prague, Czech Republic, June. Association for Computational Linguistics.

Samuel Brody and Mirella Lapata. 2008. Good Neighbors Make Good Senses: Exploiting Distributional Similarity for Unsupervised WSD. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008) - Volume 1*, pages 65–72, Manchester, United Kingdom. The Coling 2008 Organizing Committee.

Jim Cowie, Joe Guthrie, and Louise Guthrie. 1992. Lexical Disambiguation Using Simulated Annealing. In *Proceedings of the 15th Conference on Computational Linguistics (Coling 1992) - Volume 1*, pages 359–365, Nantes, France. The COLING 1992 Organizing Committee.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Alfio Gliozzo, Claudio Giuliano, and Carlo Strapparava. 2005. Domain Kernels for Word Sense Disambiguation. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 403–410, Stroudsburg, PA, USA. Association for Computational Linguistics.

Yoan Gutiérrez, Yenier Castañeda, Andy González, Rainel Estrada, Dennys D. Piug, Jose I. Abreu, Roger Pérez, Antonio Fernández Orquín, Andrés Montoyo, Rafael Muñoz, and Franc Camara. 2013. UMCC_DLSI: Reinforcing a Ranking Algorithm with Sense Frequencies and Multidimensional Semantic Resources to solve Multilingual Word Sense Disambiguation. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 241–249, Atlanta, Georgia, USA, June. Association for Computational Linguistics.

Zellig Harris. 1968. *Mathematical Structures of Language*. New York: Interscience Publishers John Wiley & Sons.

Adam Kilgarriff and Joseph Rosenzweig. 2000. Framework and Results for English SENSEVAL. *Computers and the Humanities*, 34(1-2):15–48.

Thomas K Landauer and Susan T Dumais. 1997. A Solution to Plato's Problem: The Latent Semantic Analysis Theory of Acquisition, Induction, and Representation of Knowledge. *Psychological Review*, 104(2):211–240.

Michael Lesk. 1986. Automatic Sense Disambiguation Using Machine Readable Dictionaries: How to Tell a Pine Cone from an Ice Cream Cone. In *Proceedings of the 5th Annual International Conference on Systems Documentation*, SIGDOC '86, pages 24–26, New York, NY, USA. ACM.

Linlin Li, Benjamin Roth, and Caroline Sporleder. 2010. Topic Models for Word Sense Disambiguation and Token-based Idiom Detection. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 1138–1147, Stroudsburg, PA, USA. Association for Computational Linguistics.

Will Lowe. 2001. Towards a Theory of Semantic Space. In Johanna T. Moore and Keith Stenning, editors, *Proceedings of the 23rd Conference of the Cognitive Science Society*, pages 576–581.

Steve L. Manion and Raazesh Sainudiin. 2013. DAEBAK!: Peripheral Diversity for Multilingual Word Sense Disambiguation. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 250–254, Atlanta, Georgia, USA, June. Association for Computational Linguistics.

David Martinez, Oier Lopez de Lacalle, and Eneko Agirre. 2008. On the Use of Automatically Acquired Examples for All-nouns Word Sense Disambiguation. *Journal of Artificial Intelligence Research*, 33(1):79–107, September.

Tristan Miller, Chris Biemann, Torsten Zesch, and Iryna Gurevych. 2012. Using Distributional Similarity for Lexical Expansion in Knowledge-based Word Sense Disambiguation. In *Proceedings of the 24th International Conference on Computational Linguistics (Coling 2012)*, pages 1781–1796, Mumbai, India, December. The COLING 2012 Organizing Committee.

Roberto Navigli and Mirella Lapata. 2010. An experimental study of graph connectivity for unsupervised word sense disambiguation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(4):678–692.

Roberto Navigli and Simone Paolo Ponzetto. 2012. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

Roberto Navigli, David Jurgens, and Daniele Vannella. 2013. SemEval-2013 Task 12: Multilingual Word Sense Disambiguation. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 222–231, Atlanta, Georgia, USA, June. Association for Computational Linguistics.

Simone Paolo Ponzetto and Roberto Navigli. 2010. Knowledge-rich Word Sense Disambiguation Rivaling Supervised Systems. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 1522–1531, Stroudsburg, PA, USA. Association for Computational Linguistics.

Didier Schwab, Andon Tchechmedjiev, Jérôme Goulian, Mohammad Nasiruddin, Gilles Sérasset, and Hervé Blanchon. 2013. GETALP System : Propagation of a Lesk Measure through an Ant Colony Algorithm. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 232–240, Atlanta, Georgia, USA, June. Association for Computational Linguistics.

Florentina Vasilescu, Philippe Langlais, and Guy Lapalme. 2004. Evaluating Variants of the Lesk Approach for Disambiguating Words. In *Proceedings of the 4th Conference on Language Resources and Evaluation (LREC '04)*, pages 633–636.

Tong Wang and Graeme Hirst. 2014. Applying a Naive Bayes Similarity Measure to Word Sense Disambiguation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, Baltimore, MD, USA, June.