Identifying Bad Semantic Neighbors for Improving Distributional Thesauri

Olivier Ferret
CEA, LIST, Vision and Content Engineering Laboratory,
Gif-sur-Yvette, F-91191 France.
olivier.ferret@cea.fr

Abstract

Distributional thesauri are now widely used in a large number of Natural Language Processing tasks. However, they are far from containing only interesting semantic relations. As a consequence, improving such thesaurus is an important issue that is mainly tackled indirectly through the improvement of semantic similarity measures. In this article, we propose a more direct approach focusing on the identification of the neighbors of a thesaurus entry that are not semantically linked to this entry. This identification relies on a discriminative classifier trained from unsupervised selected examples for building a distributional model of the entry in texts. Its bad neighbors are found by applying this classifier to a representative set of occurrences of each of these neighbors. We evaluate the interest of this method for a large set of English nouns with various frequencies.

1 Introduction

The work we present in this article focuses on the automatic building of a thesaurus from a corpus. As illustrated by Table 1, such thesaurus gives for each of its entries a list of words, called *semantic neighbors*, that are supposed to be semantically linked to the entry. Generally, each neighbor is associated with a weight that characterizes the strength of its link with the entry and all the neighbors of an entry are sorted according to the decreasing order of their weight.

The term *semantic neighbor* is very generic and can have two main interpretations according to the kind of semantic relations it is based on: one relies only on paradigmatic relations, such as hypernymy or synonymy, while the other considers syntagmatic relations, called collocation relations by (Halliday and Hasan, 1976) in the context of lexical cohesion or "non-classical relations" by (Morris and Hirst, 2004). The distinction between these two interpretations refers to the distinction between the notions of *semantic similarity* and *semantic relatedness* as it was done in (Budanitsky and Hirst, 2006) or in (Zesch and Gurevych, 2010) for instance. However, the limit between these two notions is sometimes hard to find in existing work as terms *semantic similarity* and *semantic relatedness* are often used interchangeably. Moreover, *semantic similarity* is frequently considered as included into *semantic relatedness* and the two problems are often tackled by using the same methods. In the remainder of this article, we will use the term *semantic similarity* with its generic sense and the term *semantic relatedness* for referring more specifically to similarity based on syntagmatic relations.

Following work such as (Grefenstette, 1994), a widespread way to build a thesaurus from a corpus is to use a semantic similarity measure for extracting the semantic neighbors of the entries of the thesaurus. Three main ways of implementing such measures can be distinguished. The first one relies on handcrafted resources in which semantic relations are clearly identified. Work based on WordNet-like lexical networks for building semantic similarity measures such as (Budanitsky and Hirst, 2006) or (Pedersen et al., 2004) falls into this category. These measures typically exploit the hierarchical structure of these networks, based on hypernymy relations. The second approach makes use of a less structured source of knowledge about words such as the definitions of classical dictionaries or the *glosses* of WordNet. WordNet's *glosses* were used to support Lesk-like measures in (Banerjee and Pedersen, 2003) and more recently, measures were also defined from Wikipedia or Wiktionaries (Gabrilovich and

Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pages 561–571,
Sofia, Bulgaria, August 4-9 2013.

Markovitch, 2007). The last option is the corpus-based approach, based on the distributional hypothesis (Firth, 1957): each word is characterized by the set of contexts from a corpus in which it appears and the semantic similarity of two words is computed from the contexts they share. This perspective was first adopted by (Grefenstette, 1994) and (Lin, 1998) and then, explored in details in (Curran and Moens, 2002b), (Weeds, 2003) or (Heylen et al., 2008).

The problem of improving the results of the "classical" implementation of the distributional approach as it can be found in (Curran and Moens, 2002a) for instance was already tackled by some work. A part of these proposals focus on the weighting of the elements that are part of the contexts of words such as (Broda et al., 2009), in which the weights of context elements are turned into ranks, or (Zhitomirsky-Geffet and Dagan, 2009), followed and extended by (Yamamoto and Asakura, 2010), that proposes a bootstrapping method for modifying the weights of context elements according to the semantic neighbors found by an initial distributional similarity measure. However, another part of these proposals implies more radical changes. The use of dimensionality reduction techniques, for instance Latent Semantic Analysis in (Padó and Lapata, 2007), the multi-prototype (Reisinger and Mooney, 2010) or examplar-based models (Erk and Pado, 2010), the Deep Learning approach of (Huang et al., 2012) or the redefinition of the distributional approach in a Bayesian framework (Kazama et al., 2010) can be classified into this second category.

The work we present in this article takes place in the framework defined by (Grefenstette, 1994) for implementing the distributional approach but proposes a new method for improving a thesaurus built in this context based on the identification of its bad semantic neighbors rather than on the adaptation of the weight of their features.

2 Principles

Our work shares with (Zhitomirsky-Geffet and Dagan, 2009) the use of a kind of bootstrapping as it starts from a distributional thesaurus and to some extent, exploits it for its improvement. However, it adopts a more indirect approach: instead of selecting the "best" semantic neighbors of an entry in the thesaurus for adapting the weights of distributional context elements, it focuses on the detection of its bad semantic neighbors, that is to say the neighbors of the entry that are actually not semantically similar to the entry. In Table 1, *waterworks* for the entry *cabdriver* and *hollowness* for the entry *machination* are two examples of such kind of neighbors. By discarding these bad neighbors or at least by downgrading them, the rank of true semantic neighbors is expected to be lower. This makes the thesaurus more interesting to use since the quality of such thesaurus strongly decreases as the rank of the neighbors of its entries increases (see Section 4.1 for an illustration), which means in practice that only the first neighbors of an entry can be generally exploited.

The approach we propose for identifying the bad semantic neighbors of a thesaurus entry relies on the distributional hypothesis, as the method for the initial building of the thesaurus, but implements it in a different way. This hypothesis roughly specifies that from a semantic viewpoint, the meaning of a word can be characterized by the set of contexts in which this word occurs. As a consequence, two words are considered as semantically similar if they occur in a large enough set of shared contexts. In work such as (Curran and Moens, 2002a), this hypothesis is implemented by collecting for each entry the words it co-occurs with in a large corpus. This co-occurrence can be based either on the position of the word in the text in relation to the entry or on the presence of a syntactic relation between the entry and the word. As a result, the distributional representation of a word takes the unstructured form of a bag of words or the more structured form of a set of pairs {syntactic relation, word}. A variant of this approach was proposed in (Kazama et al., 2010) where the distributional representation of a word is modeled as a multinomial distribution with Dirichlet as prior.

However, this approach globally faces a certain lack of diversity and complexity of the features of its models. For instance, features such as ngrams of words or ngrams of parts of speech are not considered whereas they are widely used in tasks such as word sense disambiguation (WSD) for instance, probably because they would lead to very large models and because similarity measures such as the *Cosine* measure are not necessarily suitable for heterogeneous representations (Alexandrescu and Kirchhoff, 2007). Hence, we propose in this article to build a discriminative model for repre-

abnormality	defect [0.30], disorder [0.23], deformity [0.22], mutation [0.21], prolapse [0.21], anomaly [0.21] ...
agreement	accord [0.44], deal [0.41], pact [0.38], treaty [0.36], negotiation [0.35], proposal [0.32], arrangement [0.30] ...
cabdriver	waterworks [0.23], toolmaker [0.22], weaponeer [0.17], valkyry [0.17], wang [0.17], amusement-park [0.17] ...
machination	hollowness [0.15], share-price [0.12], clockmaker [0.12], huguenot [0.12], wrangling [0.12], alternation [0.12] ...

Table 1: First neighbors of some entries of the distributional thesaurus of section 3.2

senting the contexts of a word since this kind of models are known to integrate easily a wide set of different types of features. This model aims more precisely at discriminating from a semantic viewpoint a word in context, *i.e.* in a sentence, from all other words and more particularly, from those of its neighbors in a distributional thesaurus that are likely to be actually not semantically similar to it. The underlying hypothesis follows the distributional principles: a word and a synonym should appear in the same contexts, which means that they are characterized by the same features. As a consequence, a model based on these features that can identify a word in a sentence is likely to identify also a synonym of this word in a sentence, and by extension, to identify a word that is paradigmatically linked to it. More precisely, we found that such model is specifically effective for discarding the bad neighbors of the entries of a distributional thesaurus.

3 Improving a distributional thesaurus

3.1 Overview

The principles presented in the previous section face one major problem compared to the "classical" distributional approach : the semantic similarity of two words can be evaluated directly by computing the similarity of their distributional representations. However, in our case, since this representation is a discriminative model, the similarity of two words can not be evaluated through the direct comparison of their models. These models have to be applied to words in context for being exploited. As a consequence, for deciding whether a neighbor of a thesaurus entry is a bad neighbor or not, the discriminative model of the entry has to be applied to occurrences of this neighbor in texts. Hence, the method we propose for improving a distributional thesaurus applies the following process to each of its entries:

- building of a classifier for determining whether a word in a sentence corresponds or not to the entry;
- selection of a set of examples sentences for each of the neighbors of the entry in the thesaurus;
- application of the classifier to these sentences;
- identification of bad neighbors according to the results of the classifier;
- reranking of entry's neighbors according to bad neighbors.

3.2 Building of the initial thesaurus

Before introducing our method for improving distributional thesauri, we first present the way we build such a thesaurus. As in (Lin, 1998) or (Curran and Moens, 2002a), this building is based on the definition of a semantic similarity measure from a corpus. The corpus used for defining this measure was the AQUAINT-2 corpus, a middle-size corpus made of around 380 million words coming from news articles. Although our target language is English, we chose to limit deliberately the level of the tools applied for preprocessing texts to part-of-speech tagging and lemmatization to make possible the transposition of our method to a large set of languages. This seems to be a reasonable compromise between the approach of (Freitag et al., 2005), in which none normalization of words is done, and the more widespread use of syntactic parsers in work such as (Lin, 1998). More precisely, we used *TreeTagger* (Schmid, 1994) for performing the linguistic preprocessing of the AQUAINT-2 corpus.

For the extraction of distributional data and the characteristics of the distributional similarity measure, we adopted the options of (Ferret, 2010), resulting from a kind of grid search procedure performed with the extended TOEFL test proposed in (Freitag et al., 2005) as an optimization objective. More precisely, the following characteristics were taken:

- distributional contexts made of the co-occurrents collected in a 3 word window centered on each occurrence in the corpus of the target word. These co-occurrents were restricted to nouns, verbs and adjectives;
- soft filtering of contexts: removal of co-occurrents with only one occurrence;
- weighting function of co-occurrents in con-

texts = *Pointwise Mutual Information* (PMI) between the target word and the co-occurrent;
- similarity measure between contexts, for evaluating the semantic similarity of two words = *Cosine* measure.

The building of our initial thesaurus from the similarity measure above was performed classically by extracting the closest semantic neighbors of each of its entries. More precisely, the selected measure was computed between each entry and its possible neighbors. These neighbors were then ranked in the decreasing order of the values of this measure and the first 100 neighbors were kept as the semantic neighbors of the entry. Both entries and possible neighbors were AQUAINT-2 nouns whose frequency was higher than 10.

3.3 Building a discriminative model of words in context

As mentioned in section 3.1, the starting point of our reranking process is the definition of a model for determining to what extent a word in a sentence, which is not supposed to be known in the context of this task, corresponds or not to a reference word E. This task can also be viewed as a tagging task in which the occurrences of a target word T are labeled with two tags: E and $notE$. In the context of our global objective, we are not of course interested by this task itself but rather by the fact that such classifier is likely to model the contexts in which E occurs and as a consequence, is also likely to model its meaning according to the distributional hypothesis.

A step further, such classifier can be viewed as a means for testing whether or not a word has the same meaning as E. This is a problem close to WSD as it is performed in the context of the pseudo-word disambiguation paradigm (Gale et al., 1992): a pseudo-word is created with two senses, E and $notE$, $notE$ corresponding to one or several words that are supposed to be representative of a meaning different from the meaning of E. The objective is then to build a classifier for distinguishing the pseudo-senses E and $notE$. As a consequence of this view, we adopt the same kind of features as the ones used for WSD for building our classifier. More precisely, we follow (Lee and Ng, 2002), a reference work for WSD, by adopting a Support Vector Machines (SVM) classifier with a linear kernel and three kinds of features for characterizing each considered occurrence in a text of the reference word E:

- neighboring words;
- Part-of-Speech (POS) of neighboring words;
- local collocations.

Only features based on syntactic relations are not taken from (Lee and Ng, 2002) since their use would have not been coherent with the window based approach of the building of our initial thesaurus.

For the *neighboring words* features, we consider all plain words (common and proper nouns, verbs and adjectives) and adverbs that are present in the same sentence of an occurrence of E. Each neighboring word is represented under its lemma form as a binary feature whose value is equal to 1 when it is present in the same sentence as E.

For the second type of features, we take more precisely the POS of the three words before E and those of the three words after E. Each pair {POS, position} corresponds to a binary feature for the SVM classifier. A special *empty* symbol is used for the POS when the position goes beyond the end or the beginning of the current sentence. Since we analyze texts with *TreeTagger*, the tagset is very close to the set of Penn Treebank tags.

Finally, the *local collocations* features correspond to pairs of words, named collocations, in the neighborhood of E. A collocation is specified by the notation $C_{i,j}$, with i and j referring to the position of the first and the second word of the collocation. In our case, i and j take their values in the interval $[-3, +3]$, similarly to POS. More precisely, the following 11 types of collocations are extracted for each occurrence of E: $C_{-1,-1}$, $C_{1,1}$, $C_{-2,-2}$, $C_{2,2}$ $C_{-2,-1}$, $C_{-1,1}$, $C_{1,2}$, $C_{-3,-1}$, $C_{-2,1}$, $C_{-1,2}$ and $C_{1,3}$. As for POS, a special empty symbol stands for words beyond the end or the beginning of the sentence and similarly to *neighboring words* features, words in collocations are given under their lemma form. Each instance of the 11 types of collocations is represented by a tuple ⟨lemma1, position1, lemma2, position2⟩ and leads to a binary feature for the SVM classifier.

In accordance with the process of section 3.1, a specific SVM classifier is trained for each entry of our initial thesaurus, which requires the unsupervised selection of a set of positive and negative examples. The case of positive examples is simple: a fixed number of sentences containing at least one occurrence of the target entry are randomly chosen in the corpus used for building our

initial thesaurus and the first occurrence of this entry in the sentence is taken as a positive example. Since we want to characterize words as much as possible from a semantic viewpoint, the selection of negative examples is guided by our initial thesaurus. Choosing a neighbor of the entry with a high rank would guarantee in principle few false negative examples, that is to say words[1] which are semantically similar to the entry, since the number of such neighbors strongly decreases as the rank of neighbors increases as we will illustrate it in section 4.1. In practice, taking neighbors with a rather small rank as negative examples is a better option because these examples are more useful in terms of discrimination as they are close to the transition zone between negative and positive examples. Moreover, in order to limit the risk of selecting only false negative examples, three neighbors are taken as negative examples, at ranks 10, 15 and 20[2]. For each of these negative examples, a fixed number of sentences is selected following the same principles as for positive examples, which means that on average, the number of negative examples is equal to three times the number of positive examples. This ratio reflects the fact that among the neighbors of an entry, the number of those that are semantically similar to the entry is far lower than the number of those that are not.

3.4 Identification of bad neighbors and thesaurus reranking

Once a word-in-context classifier was trained for an entry, it is used for identifying the bad neighbors of this entry, that is to say the neighbors that are not semantically similar to it. As this classifier can only be applied to words in context, a fixed number of representative occurrences have to be selected from our reference corpus for each neighbor of the entry. This selection is performed similarly to the selection of positive and negative examples in the previous section. The application of our word-in-context classifier to each of these occurrences determines whether the context of this occurrence is likely to be compatible with the context of an occurrence of the entry.

In practice, the decision of the classifier is rarely positive, which is not surprising: even if two words are semantically equivalent, each one is characterized by specific usages, especially in a given corpus, and some features of our classifier, such as the collocation features, are more likely to capture such specificities than the unigrams of "classical" distributional contexts. As a consequence, we consider that a positive outcome of our classifier is a significant hint about the presence of a word that is semantically similar to the entry and we keep a neighbor as a "good" neighbor if at least a fixed number G of its occurrences, among those selected as reference, are tagged positively by our word-in-context classifier. Conversely, a neighbor is defined as "bad" if the number of its reference occurrences tagged positively by our classifier is lower or equal to G.

The neighbors of an entry identified as bad neighbors are not fully discarded. They are rather downgraded to the end of the list of neighbors. Among the downgraded neighbors, their initial order is left unchanged. It should be noted that the word-in-context classifier is not applied to the neighbors whose occurrences are used for its training as it would frequently lead to downgrade these neighbors, which is not necessarily optimum as we chose them with a rather low rank.

4 Experiments and evaluation

4.1 Initial thesaurus evaluation

Table 2 shows the results of the evaluation of our initial thesaurus, achieved by comparing the selected semantic neighbors with two complementary reference resources: WordNet 3.0 synonyms (Miller, 1990) [W], which characterize a semantic similarity based on paradigmatic relations, and the Moby thesaurus (Ward, 1996) [M], which gathers a larger set of types of relations and is more representative of *semantic relatedness*[3]. The fourth column of Table 2, which gives the average number of synonyms and similar words in our references for the AQUAINT-2 nouns, also illustrates the difference of these two resources in terms of richness. A fusion of the two resources is also considered [WM]. As our objective is to evaluate the extracted semantic neighbors and not the ability to rebuild the reference resources, these re-

[1]More precisely, an example here is an occurrence of a word in a text but by extension, we also use the term *example* for referring to the word itself.

[2]It should be noted that these ranks come from the evaluation of section 4.1 but their choice is not the result of an optimization process.

[3]The Moby thesaurus includes more precisely both paradigmatic and syntactic relations but we will sometimes use the term *synonym* as a shortcut for referring to all the words associated to one of its entries.

freq.	ref.	#eval. words	#syn. / word	recall	R-prec.	MAP	P@1	P@5	P@10	P@100
	W	10,473	2.9	24.6	8.2	9.8	11.7	5.1	3.4	0.7
all	M	9,216	50.0	9.5	6.7	3.2	24.1	16.4	13.0	4.8
# 14,670	WM	12,243	38.7	9.8	7.7	5.6	22.5	14.1	10.8	3.8
	W	3,690	3.7	28.3	11.1	12.5	17.2	7.7	5.1	1.0
high	M	3,732	69.4	11.4	10.2	4.9	41.3	28.0	21.9	7.9
# 4,378	WM	4,164	63.2	11.5	11.0	6.5	41.3	26.8	20.8	7.3
	W	3,732	2.6	28.6	10.4	12.5	13.6	5.8	3.7	0.7
middle	M	3,306	41.3	9.3	6.5	3.1	18.7	13.1	10.4	3.8
# 5,175	WM	4,392	32.0	9.8	9.3	7.4	20.9	12.3	9.3	3.2
	W	3,051	2.3	11.9	2.1	3.3	2.6	1.2	0.9	0.3
low	M	2,178	30.1	2.8	1.2	0.5	2.5	1.5	1.5	0.9
# 5,117	WM	3,687	18.9	3.5	2.1	2.4	3.3	1.7	1.5	0.7

Table 2: Evaluation of semantic neighbor extraction

sources were filtered to discard entries and synonyms that are not part of the AQUAINT-2 vocabulary (see the difference between the number of words in the first column and the number of evaluated words of the third column). Since the frequency of words is an important factor in distributional approaches, we give our results globally but also for three ranges of frequencies that split our set of nouns into roughly equal parts: *high* frequency (frequency $>$ 1000), *middle* frequency (100 $<$ frequency $\leq$ 1000) and *low* frequency (10 $<$ frequency $\leq$ 100). These results take the form of several measures and start at the fifth column by the proportion of the synonyms and similar words of our references that are found among the first 100 extracted neighbors of each noun. As these neighbors are ranked according to their similarity value with their target word, the evaluation measures are taken from the Information Retrieval field by replacing documents with synonyms and queries with target words (see the four last columns of Table 2). The R-precision (R-prec.) is the precision after the first R neighbors were retrieved, R being the number of reference synonyms; the Mean Average Precision (MAP) is the average of the precision value after a reference synonym is found; precision at different cut-offs is given for the 1, 5, 10 and 100 first neighbors. All these values are given as percentages.

The results of Table 2 lead to three main observations. First, the level of results heavily depends on the frequency range of target words: the best results are obtained for high frequency words while evaluation measures significantly decrease for words whose frequency is low. Second, the characteristics of the reference resources have a significant impact on results. WordNet provides a restricted number of synonyms for each noun while the Moby thesaurus contains for each entry a large number of synonyms and similar words. As a consequence, the precisions at different cut-offs have a significantly higher value with Moby as reference than with WordNet as reference. Finally, the results of Table 2 are compatible with those of (Lin, 1998) for instance (R-prec. = 11.6 and MAP = 8.1 with WM as reference for all entries of the thesaurus at `http://webdocs.cs.ualberta.ca/lindek/Downloads/sim.tgz`) if we take into account the fact that the thesaurus of Lin was built from a much larger corpus and with syntactic co-occurrences.

4.2 Implementation issues

The implementation of the method we have presented in section 3 raises several issues. One of these concerns the occurrences to select from texts of both the entries of the thesaurus and their neighbors. These occurrences are used both for the training of our word-in-context classifier and for the identification of bad neighbors. In practice, we extract randomly from our reference corpus, *i.e.* the AQUAINT-2 corpus, a fixed number of sentences, equal to 250, for each word of the vocabulary of our initial thesaurus and exploit them for the two tasks. This extraction is performed on the basis of the lemma form of these words. It should be noted that 250 is the upper limit of the number of occurrences by word since the frequency in the corpus of many words is lower than 250. When this limit is not reached, all the available oc-

currences are taken, which may be no more than 11 occurrences for certain low-frequency words. The upper limit of 250 is halfway between the 385 training examples on average for the Lexical Sample Task of Senseval 1 and the 118 training examples on average for the same task of Senseval 2.

The training of our word-in-context classifier is also an important issue. As mentioned before, this classifier is a linear SVM. Hence, only its C regularization parameter can be optimized. Since we have one specific classifier for each thesaurus entry, such optimization has globally a high cost, even for a linear kernel. Hence, we have first evaluated through a 5-fold cross-validation method the results of these classifiers with a default value of C, equal to 1. Table 3 gives their average accuracy value along with their standard deviation for all the entries of the thesaurus and for the three frequency ranges of Table 2.

	all	high	middle	low
accuracy	86.2	86.1	86.0	86.5
standard deviation	6.1	4.2	5.7	7.6

Table 3: Results of word-in-context classifiers

This table shows a global high level of result along with similar values for all the frequency ranges of entries[4]. Hence, we have decided not to optimize the C parameter and to adopt the default value of 1 for all the word-in-context classifiers.

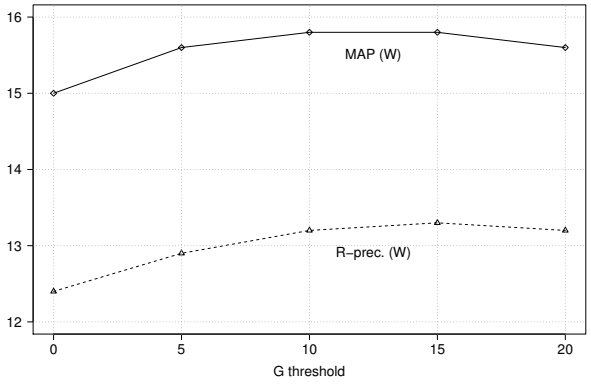

Figure 1: R-precision and MAP for various values of the G threshold

The last and the most important implementation issue is the setting of the threshold G for determining whether a neighbor is likely to be a bad neighbor. For this setting, we have randomly chosen a subset of 859 entries of our initial thesaurus that corresponds to 10% of the entries with at least one true neighbor in any of our references. Figure 1 gives the results of the reranked thesaurus for these entries in terms of R-precision and MAP against reference W[5] for various values of G. Although the level of these measures does not change a lot for $G > 5$, the graph of Figure 1 shows that $G = 15$ appears to be an optimal value. Hence, this is the value used for the detailed evaluation of the next section.

4.3 Evaluation of the reranked thesaurus

Table 4 gives the evaluation of the application of our reranking method to the initial thesaurus according to the same principles as in section 4.1. The value of each measure comes with its difference with the corresponding value for the initial thesaurus. As the recall measure and the precision for the last rank do not change in a reranking process, they are not given again.

The first thing to notice is that at the global scale, all measures for all references are significantly improved[6], which means that our hypothesis about the possibility for a discriminative classifier to capture the meaning of a word tends to be validated. It is an interesting result since the features upon which this classifier was built were taken from WSD and were not specifically selected for this task. As a consequence, there is probably some room for improvement.

If we go into details, Table 4 clearly shows two main trends. First, the improvement of results is particularly effective for middle frequency entries, then for low frequency and finally, for high frequency entries. Because of their already high level in the initial thesaurus, results for high frequency entries are difficult to improve but it is important to note that our selection of bad neighbors has a very low error rate, which at least preserves these results. This is confirmed by the fact that, with WordNet as reference, only 744 neighbors were found wrongly downgraded, spread over 686 entries, which represents only 5% of all downgraded neighbors. The second main trend of Table 4 con-

[4]The standard deviation is a little bit higher for the lowest frequencies but it should be noted that the low number of examples for low frequency entries does not seem to have a strong impact on the results of such classifier.

[5]The use of W as reference is justified by the fact that the number of synonyms for an entry in W is more compatible, especially for R-precision, with the real use of the resulting thesaurus in an application.

[6]The statistical significance of differences with the initial thesaurus was evaluated by a paired Wilcoxon test with p-value < 0.05 and < 0.01 († and ‡ for non significance).

freq.	ref.	R-prec.		MAP		P@1		P@5		P@10	
all	W	9.1	(0.9)	10.7	(0.9)	12.8	(1.1)	5.6	(0.5)	3.7	(0.3)
	M	7.2	(0.5)	3.5	(0.3)	26.5	(2.4)	17.9	(1.5)	14.0	(1.0)
	WM	8.4	(0.7)	6.1	(0.5)	24.8	(2.3)	15.4	(1.3)	11.7	(0.9)
high	W	11.3	(0.2) †	12.6	(0.1)	17.3	(0.1) ‡	7.8	(0.1) ‡	5.1	(0.0)
	M	10.3	(0.1)	4.9	(0.0)	42.1	(0.8)	28.4	(0.4)	22.1	(0.2)
	WM	11.1	(0.1)	6.6	(0.1)	42.0	(0.7)	27.2	(0.4)	20.9	(0.1)
middle	W	11.8	(1.4)	13.8	(1.3)	15.7	(2.1)	6.5	(0.7)	4.1	(0.4)
	M	7.3	(0.8)	3.6	(0.5)	23.3	(4.6)	16.0	(2.9)	12.4	(2.0)
	WM	10.3	(1.0)	8.1	(0.7)	25.1	(4.2)	14.6	(2.3)	10.9	(1.6)
low	W	3.2	(1.1)	4.6	(1.3)	3.9	(1.3)	1.8	(0.6)	1.3	(0.4)
	M	1.8	(0.6)	0.8	(0.3)	4.4	(1.9)	2.9	(1.4)	2.6	(1.1)
	WM	3.1	(1.0)	3.3	(0.9)	5.1	(1.8)	2.9	(1.2)	2.3	(0.8)

Table 4: Results of the reranking of semantic neighbors

cerns the type of semantic relations: results with Moby as reference are improved in a larger extent than results with WordNet as reference. This suggests that our procedure is more effective for semantically related words than for semantically similar words, which can be considered as a little bit surprising since the notion of context in our discriminative classifier seems *a priori* more strict than in "classical" distributional contexts. However, this point must be investigated further as a significant part of the relations in Moby, even if they do no represent the largest part of them, are paradigmatic relations.

WordNet	respect, admiration, regard
Moby	admiration, appreciation, acceptance, dignity, regard, respect, account, adherence, consideration, estimate, estimation, fame, greatness, reverence + 79 words more
initial	cordiality, gratitude, **admiration**, comradeship, back-scratching, perplexity, **respect**, ruination, appreciation, neighbourliness ...
reranking	gratitude, **admiration**, **respect**, appreciation, neighborliness, trust, empathy, goodwill, reciprocity, half-staff, affection, self-esteem, reverence, longing, **regard** ...

Table 5: Impact of our reranking for the entry *esteem*

Table 5 illustrates more precisely the impact of our reranking procedure for the middle frequency entry *esteem*. Its **WordNet** row gives all the reference synonyms for this entry in WordNet while its Moby row gives the first reference related words for this entry in Moby. In our *initial* thesaurus, the first two neighbors of *esteem* that are present in our reference resources are *admiration* (rank 3) and *respect* (rank 7). The reranking produces a thesaurus in which these two words appear as the second and the third neighbors of the entry because neighbors without clear relation with it such as *back-scratching* were downgraded while its third synonym in WordNet is raised from rank 22 to rank 15. Moreover, the number of neighbors among the first 15 ones that are present in Moby increases from 3 to 5.

5 Related work

The building of distributional thesaurus is generally viewed as an application or a mode of evaluation of work about semantic similarity or semantic relatedness. As a consequence, the improvement of such thesaurus is generally not directly addressed but is a possible consequence of the improvement of semantic similarity measures. However, the extent of this improvement is rarely evaluated as most of the work about semantic similarity is evaluated on datasets such as the WordSim-353 test collection (Gabrilovich and Markovitch, 2007), which are only partially representative of the results for thesaurus building.

If we consider more specifically the problem of improving semantic similarity, and by the way thesauri, in a given paradigm, (Broda et al., 2009), (Zhitomirsky-Geffet and Dagan, 2009) and (Yamamoto and Asakura, 2010), which all take place in the paradigm defined by (Grefenstette, 1994), are the closest works to ours. (Broda et al., 2009) proposes a new weighting scheme of words in distributional contexts that replaces the weight of

word by a function of its rank in the context, which is a way to be less dependent on the values of a particular weighting function. (Zhitomirsky-Geffet and Dagan, 2009) shares with our work the use of bootstrapping by relying on an initial thesaurus to derive means of improving it. More specifically, (Zhitomirsky-Geffet and Dagan, 2009) assumes that the first neighbors of an entry are more relevant than the others and as a consequence, that their most significant features are also representative of the meaning of the entry. The neighbors of the entry are reranked according to this hypothesis by increasing the weight of these features to favor their influence in the distributional contexts that support the evaluation of the similarity between the entry and its neighbors. (Yamamoto and Asakura, 2010) is a variant of (Zhitomirsky-Geffet and Dagan, 2009) that takes into account a larger number of features for the reranking process. One main difference between all these works and ours is that they assume that the initial thesaurus was built by relying on distributional contexts represented as bags-of-words. Our method does not make this assumption as its reranking is based on a classifier built in an unsupervised way[7] from and applied to the corpus used for building the initial thesaurus. As a consequence, it could even be applied to other paradigms than (Grefenstette, 1994).

If we focus more specifically on the improvement of distributional thesauri, (Ferret, 2012) is the most comparable work to ours, both because it is specifically focused on this task and it is based on the same evaluation framework. (Ferret, 2012) selects in an unsupervised way a set of positive and negative examples of semantically similar words from the initial thesaurus, uses them for training a classifier deciding whether or not a pair of words are semantically similar and finally, applies this classifier to the neighbors of each entry for reranking them. One of the objectives of (Ferret, 2012) was to rebalance the initial thesaurus in favor of low frequency entries. Although this objective was reached, the resulting thesaurus tends to have a lower performance than the initial thesaurus for high frequency entries and for synonyms. The problem with high frequency entries comes from the fact that applying a machine learning classifier to its training examples does not lead to a perfect result. The problem with synonyms arises from the imbalance between *semantic similarity* and *semantic relatedness* among training examples: most of selected examples were pairs of words linked by *semantic relatedness* because this kind of relations are more frequent among semantic neighbors than relations based on *semantic similarity*.

In both cases, the method proposed in (Ferret, 2012) faces the problem of relying only on the distributional thesaurus it tries to improve. This is an important difference with the method presented in this article, which mainly exploits the context of the occurrences of words in the corpus used for the building the initial thesaurus. As a consequence, at a global scale, our reranked thesaurus outperforms the final thesaurus of (Ferret, 2012) for nearly all measures. The only exceptions are the P@1 values for M and WM as reference. However, it should be noted that values for both MAP and R-precision, which are more reliable measures than P@1, are identical for the two thesauri and the same references.

6 Conclusion and perspectives

In this article, we have presented a new approach for reranking the semantic neighbors of a distributional thesaurus. This approach relies on the unsupervised building of discriminative classifiers dedicated to the identification of its entries in texts, with the objective to characterize their meaning according to the distributional hypothesis. The classifier built for an entry is then applied to a set of occurrences of its neighbors for identifying and downgrading those that are not semantically related to the entry. The proposed method was tested on a large thesaurus of nouns for English and led to a significant improvement of this thesaurus, especially for middle and low frequency entries and for semantic relatedness. We plan to extend this work by taking into account the notion of word sense as it is done in (Reisinger and Mooney, 2010) or (Huang et al., 2012): since we rely on occurrences of words in texts, this extension should be quite straightforward by turning our word-in-context classifiers into true word sense classifiers.

Acknowledgments

This work was partly supported by the project ANR ASFALDA ANR-12-CORD-0023.

[7] It is a supervised classifier but its training set is selected in an unsupervised way.

References

Andrei Alexandrescu and Katrin Kirchhoff. 2007. Data-driven graph construction for semi-supervised graph-based learning in NLP. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics (NAACL HLT 2007)*, pages 204–211, Rochester, New York.

Satanjeev Bano Banerjee and Ted Pedersen. 2003. Extended gloss overlaps as a measure of semantic relatedness. In *Eighteenth International Conference on Artificial Intelligence (IJCAI-03)*, Acapulco, Mexico.

Bartosz Broda, Maciej Piasecki, and Stan Szpakowicz. 2009. Rank-Based Transformation in Measuring Semantic Relatedness. In *22nd Canadian Conference on Artificial Intelligence*, pages 187–190.

Alexander Budanitsky and Graeme Hirst. 2006. Evaluating wordnet-based measures of lexical semantic relatedness. *Computational Linguistics*, 32(1):13–47.

James Curran and Marc Moens. 2002a. Scaling context space. In *40th Annual Meeting of the Association for Computational Linguistics (ACL-02)*, pages 231–238, Philadelphia, Pennsylvania, USA.

James R. Curran and Marc Moens. 2002b. Improvements in automatic thesaurus extraction. In *Workshop of the ACL Special Interest Group on the Lexicon (SIGLEX)*, pages 59–66, Philadelphia, USA.

Katrin Erk and Sebastian Pado. 2010. Exemplar-based models for word meaning in context. In *48th Annual Meeting of the Association for Computational Linguistics (ACL 2010), short paper*, pages 92–97, Uppsala, Sweden, July.

Olivier Ferret. 2010. Testing semantic similarity measures for extracting synonyms from a corpus. In *Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta.

Olivier Ferret. 2012. Combining bootstrapping and feature selection for improving a distributional thesaurus. In *20th European Conference on Artificial Intelligence (ECAI 2012)*, pages 336–341, Montpellier, France.

John R. Firth, 1957. *Studies in Linguistic Analysis*, chapter A synopsis of linguistic theory 1930-1955, pages 1–32. Blackwell, Oxford.

Dayne Freitag, Matthias Blume, John Byrnes, Edmond Chow, Sadik Kapadia, Richard Rohwer, and Zhiqiang Wang. 2005. New experiments in distributional representations of synonymy. In *Ninth Conference on Computational Natural Language Learning (CoNLL)*, pages 25–32, Ann Arbor, Michigan, USA.

Evgeniy Gabrilovich and Shaul Markovitch. 2007. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *20th International Joint Conference on Artificial Intelligence (IJCAI 2007)*, pages 6–12.

William A Gale, Kenneth W Church, and David Yarowsky. 1992. Work on statistical methods for word sense disambiguation. In *AAAI Fall Symposium on Probabilistic Approaches to Natural Language*, pages 54–60.

Gregory Grefenstette. 1994. *Explorations in automatic thesaurus discovery*. Kluwer Academic Publishers.

Michael A. K. Halliday and Ruqaiya Hasan. 1976. *Cohesion in English*. Longman, London.

Kris Heylen, Yves Peirsmany, Dirk Geeraerts, and Dirk Speelman. 2008. Modelling Word Similarity: An Evaluation of Automatic Synonymy Extraction Algorithms. In *Sixth conference on International Language Resources and Evaluation (LREC 2008)*, Marrakech, Morocco.

Eric H. Huang, Richard Socher, Christopher D. Manning, and Andrew Y. Ng. 2012. Improving word representations via global context and multiple word prototypes. In *50th Annual Meeting of the Association for Computational Linguistics (ACL'12)*, pages 873–882.

Jun'ichi Kazama, Stijn De Saeger, Kow Kuroda, Masaki Murata, and Kentaro Torisawa. 2010. A bayesian method for robust estimation of distributional similarities. In *48th Annual Meeting of the Association for Computational Linguistics*, pages 247–256, Uppsala, Sweden.

Yoong Keok Lee and Hwee Tou Ng. 2002. An empirical evaluation of knowledge sources and learning algorithms for word sense disambiguation. In *2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pages 41–48.

Dekang Lin. 1998. Automatic retrieval and clustering of similar words. In *17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics (ACL-COLING'98)*, pages 768–774, Montral, Canada.

George A. Miller. 1990. WordNet: An On-Line Lexical Database. *International Journal of Lexicography*, 3(4).

Jane Morris and Graeme Hirst. 2004. Non-classical lexical semantic relations. In *Workshop on Computational Lexical Semantics of Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 46–51, Boston, MA.

Sebastian Padó and Mirella Lapata. 2007. Dependency-based construction of semantic space models. *Computational Linguistics*, 33(2):161–199.

Ted Pedersen, Siddharth Patwardhan, and Jason Michelizzi. 2004. Wordnet::similarity - measuring the relatedness of concepts. In *HLT-NAACL 2004, demonstration papers*, pages 38–41, Boston, Massachusetts, USA.

Joseph Reisinger and Raymond J. Mooney. 2010. Multi-prototype vector-space models of word meaning. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL 2010)*, pages 109–117, Los Angeles, California, June.

Helmut Schmid. 1994. Probabilistic part-of-speech tagging using decision trees. In *International Conference on New Methods in Language Processing*.

Grady Ward. 1996. Moby thesaurus. Moby Project.

Julie Weeds. 2003. *Measures and Applications of Lexical Distributional Similarity*. Ph.D. thesis, Department of Informatics, University of Sussex.

Kazuhide Yamamoto and Takeshi Asakura. 2010. Even unassociated features can improve lexical distributional similarity. In *Second Workshop on NLP Challenges in the Information Explosion Era (NLPIX 2010)*, pages 32–39, Beijing, China.

Torsten Zesch and Iryna Gurevych. 2010. Wisdom of crowds versus wisdom of linguists - measuring the semantic relatdness of words. *Natural Language Engineering*, 16(1):25–59.

Maayan Zhitomirsky-Geffet and Ido Dagan. 2009. Bootstrapping Distributional Feature Vector Quality. *Computational Linguistics*, 35(3):435–461.