Supervised All-Words Lexical Substitution using Delexicalized Features

György Szarvas[1] **Chris Biemann**[2] **Iryna Gurevych**[3,4]
(1) Nuance Communications Deutschland GmbH
Kackertstrasse 10, D-52072 Aachen, Germany
(2) FG Language Technology
Department of Computer Science, Technische Universität Darmstadt
(3) Ubiquitous Knowledge Processing Lab (UKP-TUDA)
Department of Computer Science, Technische Universität Darmstadt
(4) Ubiquitous Knowledge Processing Lab (UKP-DIPF)
German Institute for Educational Research and Educational Information
http://www.nuance.com , http://www.ukp.tu-darmstadt.de

Abstract

We propose a supervised lexical substitution system that does not use separate classifiers per word and is therefore applicable to any word in the vocabulary. Instead of learning word-specific substitution patterns, a global model for lexical substitution is trained on delexicalized (i.e., non lexical) features, which allows to exploit the power of supervised methods while being able to generalize beyond target words in the training set. This way, our approach remains technically straightforward, provides better performance and similar coverage in comparison to unsupervised approaches. Using features from lexical resources, as well as a variety of features computed from large corpora (n-gram counts, distributional similarity) and a ranking method based on the posterior probabilities obtained from a Maximum Entropy classifier, we improve over the state of the art in the LexSub Best-Precision metric and the Generalized Average Precision measure. Robustness of our approach is demonstrated by evaluating it successfully on two different datasets.

1 Introduction

In recent years, the task of automatically providing lexical substitutions in context (McCarthy and Navigli, 2007) received much attention. The premise to be able to replace words in a sentence without changing its meaning gave rise to applications like linguistic steganography (Topkara et al., 2006; Chang and Clark, 2010), semantic text similarity (Agirre et al., 2012), and plagiarism detection (Gipp et al., 2011).

Lexical substitution, a special form of contextual paraphrasing where only a single word is replaced, is closely related to word sense disambiguation (WSD): polysemous words have possible substitutions reflecting several senses, and the correct sense has to be picked to avoid spurious system behavior. However, no explicit word sense inventory is required for lexical substitution (Dagan et al., 2006).

The prominent tasks in a lexical substitution system are *generation* and *ranking*, i.e. to generate a set of possible substitutions for the target word and then to rank this set of possible substitutions according to their contextual fitness. The task to generate a high quality set of possible substitutions is challenging in itself, for two reasons. First, the available lexical resources are seldom complete in listing synonyms. Second, manually annotated substitutions show that not all synonyms of a word are appropriate in a given context, and many good substitutions have other lexical relation than synonymy to the original word.

In this work, we present a supervised lexical substitution system that, unlike the usual *lexical sample* supervised approaches, can produce substitutions for targets that are not contained in the training material. We reach this by using non-lexical features from heterogeneous evidence, including lexical-semantic resources and distributional similarity, n-gram and shallow syntactic features based on large, unannotated background corpora. In light of the existence of lexical resources such as WordNet (Fellbaum, 1998) or machine readable dictionaries that can serve as the source for lexical information, and with the ever-increasing availability of large unannotated corpora for many languages and

Proceedings of NAACL-HLT 2013, pages 1131–1141,
Atlanta, Georgia, 9–14 June 2013. ©2013 Association for Computational Linguistics

domains, our proposal enables us to leverage the quality gain of supervised machine learning while generalizing over a large vocabulary through the avoidance of lexicalized features. Using a single classifier for all substitution targets in this way results in an all-words substitution system. As our results demonstrate, our model improves over the state of the art in lexical substitution with practically no open parameters that have to be optimized and selected carefully according to the dataset at hand.

2 Related Work

Previous works in lexical substitution either address both the *generation* and the *ranking* tasks, and are therefore applicable to any word without prelabeled data (c.f. the Semeval 2007 task (McCarthy and Navigli, 2007) and related work) or focus on the more challenging ranking step only (c.f. Erk and Padó (2008) and related work). The latter approaches take the list of possible substitutions directly from the testing data as a workaround to generating the possible substitutions, and merely evaluate the ranking capabilities of these methods.

The most accurate lexical substitution systems use supervised machine learning to train (and test) a separate classifier per target word, using lexical and shallow syntactic features. These systems rely on the existence of a large number of annotated examples (i.e. sentences together with the contextually valid substitutions) for each word. Biemann (2012) describes a supervised lexical substitution system for frequent nouns. Exploiting a large amount of sense tagged examples and (sense-specific) data annotated with substitutions, an accurate coarse-grained WSD model is trained and then the most frequent substitutions of the predicted sense are assigned to the new occurrences of the target words. The results demonstrate that lexical substitution of noun targets can be attained with very high precision (over 90%) if sufficient training material is available. However, due to high annotation costs, methods that do not require labeled training data per target scale better to a large vocabulary.

Knowledge-based systems like e.g. by Hassan et al. (2007), who use a number of knowledge-based and unsupervised methods and combine these clues using a voting scheme, do not need training data per target. The combination of different signals, however, has to be done manually. Unsupervised systems that rely on distributional similarity (Thater et al., 2011) or topic models (Li et al., 2010) are single signals in this sense, and their development is guided by the performance and observations on standard datasets. Such signals, however, can also be kept simple avoiding any task-specific optimization and can be integrated in a single model for all words using a limited amount of training data and delexicalized features, as in Senselearner (Mihalcea and Csomai, 2005) for weakly supervised all-words disambiguation. This way, task specific development can be replaced by a machine learning component and the resulting model applies also to unseen words, similar to the knowledge-based approaches.

2.1 Full Lexical Substitution Systems

Related works that address the lexical substitution problem according to the settings established by the *English Lexical Substitution Task* (McCarthy and Navigli, 2007) at Semeval 2007 (LexSub) typically employ a simple ranking strategy based on local n-gram frequencies and focus on finding an optimal source of possible substitutions, as the selection of lexical resources has largest impact on the overall system performance: Sinha and Mihalcea (2009) systematically explored the benefits of multiple lexical resources and found that a supervised combination of several resources lead to statistically significant improvements in accuracy (about 3.5% points over the best single resource, WordNet). They tested LSA (Deerwester et al., 1990), ESA (Gabrilovich and Markovitch, 2007) and n-gram frequencies for contextualization and found n-gram frequencies to be more effective than dimensionality reduction techniques by a large margin. Their improvements were obtained by supervised learning on the *combination* of several lexical resources. Our work, on the other hand, is concerned with using more advanced features and we obtain significant improvements based on a diverse set of features and a different learning setup: we train a *model for contextualization*, rather than to combine substitutions from several different resources.

A recent work by Sinha and Mihalcea (2011) used an approach based on graph centrality to rank the candidates and achieved comparable performance to n-gram-frequency-based ranking. To summarize, the use of n-gram frequencies for ranking and WordNet as the (most appropriate single) source of synonyms is competitive to more complex solutions and provides a simple and strong lexical substitution system. This motivated the follow-up work by Chang and Clark (2010) to use WordNet and n-grams in a linguistic steganography application and this motivates us to use this method as our baseline.

2.2 Ranking Word Meaning in Context

Another prominent line of related work focused solely on the accurate ranking of a pre-given set of possible synonyms, according to their plausibility as a substitution in a given context. Typically, lexical substitution data is used for evaluation purposes, taking the candidate substitutions directly from the test data. This choice is motivated by the assumption that better semantic models should rank near-synonyms more accurately according to how they fit in the original word's context.

Erk and Padó (2008) proposed the use of multiple vector representations of words, where the basic representation corresponds to a standard co-occurrence vector, while further vectors are used to characterize words according to their inverse selectional preference statistics for typical dependency relations. The representation of a word in its context is computed via combining the basic representation of a word with the inverse selectional preference vectors of its related words from the context. Ranking is done by comparing vectors of possible substitutions with the substitution target. Thater et al. (2010) took a similar approach but used second order co-occurrence vectors and report improved performance.

An exemplar-based approach is presented by Erk and Padó (2010) and Reisinger and Mooney (2010b) to model word meaning with respect to its context: instead of representing the word and the context as separate vectors and combining them, a set of word occurrences in similar contexts is picked first, and then only these exemplars are used to represent the word in context. While this approach provides good results with relatively simple and transparent models, each occurrence of a word has a unique representation (that can only be computed at testing time), and it is computationally expensive to scale these models to a large number of examples.

Dinu and Lapata (2010) used a bag of words latent variable model to characterize the meaning of a word as a distribution over a set of latent variables (that is, probabilistic senses). Contextualized representation of word meaning is then attained by conditioning the model on the context words in which the target word occurs. A similar approach has been evaluated for word similarity (Reisinger and Mooney, 2010a) and word sense disambiguation (Li et al., 2010).

Although our main goal here is to develop a full-fledged lexical substitution system, we mainly focus on the construction of better ranking models based on supervised machine learning and delexicalized features that scale well for unseen words. This approach has similar properties (applicability to all words without word-specific training data) to the knowledge-based and unsupervised models described above, so we will also refer to these systems for comparison.

3 Datasets

In our work, we use two major freely available datasets that contain human-annotated substitutions for single words in their full-sentence context.

3.1 LexSub dataset

This dataset was introduced in the Lexical Substitution task at Semeval 2007[1]. It consists of 2002 sentences for a total of 201 words (10 sentences per word, but 8 sentences does not have gold standard labels). Each sentence was assigned to 5 native speaker annotators, who entered as many paraphrases or substitutions as they found appropriate for the word in context. Paraphrases are assigned a weight (or frequency) that denotes how many annotators suggested that particular word as a substitute.

3.2 TWSI

A similar, but larger dataset is the Turk Bootstrap Word Sense Inventory (TWSI[2], (Biemann, 2012)). The data was collected through a three-step crowdsourcing process and comprises 24,647 sentences

[1] download at http://nlp.cs.swarthmore.edu/semeval/tasks/task10/data.shtml

[2] http://www.ukp.tu-darmstadt.de/data/lexical-resources/twsi-lexical-substitutions/

for a total of 1,012 target nouns, where crowdworkers have provided substitutions for a target word in context. We did not use the roughly 150,000 sense-labeled contexts and the sense inventory of this resource, i.e. this dataset – as used in this study – is transparent to the LexSub data. For the majority of the data, responses from 3 annotators were collected per context, and there are on average 24 sentences per target word in the dataset. Due to this, the average weight of good substitutions is somewhat lower than in the LexSub dataset (1.27 vs. 1.58 in LexSub), but the average number of unique substitutions per target word is slightly higher in TWSI (average of 22 words / target vs. 17 in LexSub).

3.3 Source of Possible Substitutions

In our lexical substitution system, we used WordNet as the source for candidate synonyms. For each substitution target, we took all synonyms from all of the word's WordNet synsets as candidates, together with the words from synsets in *similar to, entailment* and *also see* relation to these synsets[3]. In order to evaluate and compare our ranking methodology in a transparent way with those studies that focused just on the candidate ranking task, we also performed experiments where we pooled the set of candidates from the gold standard dataset. This setting ensures that each set contains a positive candidate, and that all human-suggested paraphrases are available as positive examples for a given sentence.

The main characteristics of the datasets (with both WordNet or the gold standard as the source of candidate substitutions) are summarized in Table 1. The rows in the table indicate the source of possible substitutions, number of target words, instances with at least one non-multiword possible substitution, average size of candidate sets, and number of instances with no good candidate and frequency of different labels. The labels denote how many annotators proposed a particular word as substitution in the given context and can be interpreted as a measure of goodness: the higher the value, the better the candidate fits in the context. Similarly, the label 0 denotes the total number of negative examples in our datasets, i.e. bad substitutions – words that belong to the candidate set for a particular target word, but are not listed as good substitutions in the given context in the dataset.

	LexSub		TWSI	
source	WN	Gold St.	WN	Gold St.
# words	201	201	908	1007
#inst	2002	2002	22543	24643
avg. set	21	17	7.5	22
# empty	508	17	11165	620
#0	39465	27300	151538	443993
#1	1302	4698	10678	77417
#2	582	1251	4171	17585
#3	308	571	2069	5629
#4	212	319	74	325
#5+	129	179	121	411

Table 1: Details of the datasets: WN=WordNet

4 Methodology

4.1 Experimental Setup and Evaluation

We follow previous works in lexical substitution and evaluate our models using the Generalized Average Precision (GAP) (Kishida, 2005) measure which assesses the quality of the entire ranked list. In addition, we also provide the precision of our system at the first rank (P@1), i.e. the percentage of correct paraphrases at rank 1. This is a realistic evaluation criterion for many applications, such as paraphrasing for linguistic steganography: it is the highest-ranked candidate that can be used to replace the original word (the manipulated text should preserve the original meaning) and there is no straightforward way to exploit multiple correct answers. In addition, we also provide the Semeval 2007 *best precision*[4] metric (McCarthy and Navigli, 2007) for the LexSub dataset for comparison to Semeval 2007 participants. This metric also evaluates the first guess of a system (per context), but gives less credit to easier contexts, where several good options exist. This fact motivates us to use P@1 rather than the *best precision* metric in all other experiments.

[3] This candidate set was found best for WordNet by Martinez et al. (2007).

[4] Since our system always provides an answer, the Semeval 2007 *best recall* equals *best precision*.

4.2 Machine Learning on Delexicalized Features

After the list of potential substitutions is obtained, lexical substitution is cast as a ranking task where the goal is to prefer contextually plausible substitutions over implausible ones. The goal of this study is to learn a ranking model that is applicable to any word, for which a list of synonyms is available. A supervised model can generalize over the example target words in the datasets, if aggregate features can be defined that have the same semantics regardless of the actual context, target word or candidate substitution they are computed from. Having such a representation, one can expect to learn patterns that generalize over the words/contexts seen in the training dataset, and thus the setup constitutes a supervised all-word system.

To simulate an all-word scenario, we perform a 10-fold cross validation in our experiments, splitting the dataset into equal-sized folds randomly on the *target word* level. That is, all sentences for a particular target word fall into the same fold and thus either the training or the test set (but never both). This way we always train and test the model on disjoint sets of words and as such, the learnt models cannot exploit word-specific properties. This makes our results realistic estimates of an open vocabulary paraphrasing system, where we would apply the models (mostly) to words that were not in the training material.

4.2.1 Machine Learning Model

In our experiments, we used a Maximum Entropy (MaxEnt) classifier model implemented in the Mallet (McCallum, 2002) package and trained a binary classifier to predict if a given substitution is valid in a particular context or not.

We chose to use Maximum Entropy models for two main reasons: MaxEnt is not sensitive to parameter settings and handles correlated features well, which is crucial in our situation where many features are highly correlated.

Due to the low number of positive examples in the datasets (see Table 1, labels 1-5+) and to emphasize better paraphrases suggested by several annotators, we assigned a weight to positive instances during the training process equal to their score (the number of annotators suggesting that paraphrase; the weight of negative instances was set to 1).

The output of the MaxEnt classifier is a posterior probability distribution for each target/substitution pair, denoting the probabilities of the instance to be a good or a bad substitution, given the feature values that describe both the words and their context. The ranking over a set of candidates can be naturally induced based on their posterior scores for the positive class, i.e. a number that denotes 'how good the candidate is, given the context'. That is, the best substitution candidate s (characterized by a set of features $\mathbf{F}$) from a set of candidates $\mathbf{S}$ is obtained as $argmax_{s \in \mathbf{S}}[P(good|\mathbf{F})]$, the next best as the $argmax$ of the remaining elements, and so on.

This pointwise approach to subset ranking (Cossock and Zhang, 2008) is arguably simplistic, but several studies (c.f. Li et al. (2007; Busa-Fekete et al. (2011)) found this approach to perform reasonably well given that the model provides accurate probability estimates, which is the case for MaxEnt.

4.3 Delexicalized Features

We use heterogeneous sources of information to describe each target word/candidate substitution pair in its context. The most important features describe the syntagmatic coherence of the substitute in context, measured as local n-gram frequencies obtained from web data, in a sliding window around the target word. In addition we use features to describe the (non-positional, i.e. non-local) distributional similarity of the target and its candidate substitution in terms of sentence level co-occurrence statistics collected from newspaper texts. A further set of features captures the properties of the target and candidate word in a lexical resource (WordNet), such as their number of senses, how frequent senses are synonymous, etc. Lastly, we use part of speech patterns to describe the target word in context. This way, unlike many other methods suggested in previous works (Thater et al., 2011; Erk and Padó, 2008), our model does not require deep syntactic analysis of the test sentences in order to rank the candidates. Even though we make intensive use of WordNet to compute some of our feature functions, this is not a severe restriction for a practical paraphrasing system: one has to have a decent lexical resource in order to mine a reasonable set of candidate synonyms and such a resource can also serve as a source for features in the classifier. The rest of the feature functions exploit only large unannotated corpora and a POS tagger at application time.

For a target word *t*, and candidate substitution s_i from a set of candidates ***S***, we used the features below. Each numeric feature is used both in the form given below, and set-wise scaled to $[0, 1]$ (we leave it to the classifier to pick the more useful form of information). For the LexSub dataset, each feature is defined once for all instances, and once specific to the four POS categories in the dataset. That is each instance would have the described features defined twice, once the general form (defined for every instance) and once the form according to the *predicted* POS category of the target word. This allows the model to learn general and also POS-specific patterns based on the information described below (i.e. frequency thresholds, distributional properties etc. for nouns or verbs etc. in particular). We denote the left and right contexts around t and all words in the sentence except t with c_l, c_r and c, respectively)

4.3.1 Lexical Resource Features

We used Wordnet 3.0 as the source for substitution candidates and as a source for delexicalized features. We found the measure of ambiguity and the sense number to provide useful information in a more general context: it is informative how many senses a word has, and it is informative from which sense number of the substitution target the substitution candidate came from, since they are ordered by corpus frequency. In addition, we used the synsets IDs of the words' hypernyms as features, which can capture more general semantics (the word to replace is 'animate', 'abstract', etc.). The following features were extracted from WordNet:

- number of senses of t and s_i in WordNet
- the sense numbers of t and s_i which are synonymous (in case they are direct synonyms, c.f. WN sense numbers encode sense frequencies)
- binary features for synset IDs of the hypernyms of the synset containing t and s_i (this feature type did not significantly improve results)

4.3.2 Corpus-based Features

In order to create a Distributional Thesaurus (DT) similar to Lin (1998), we parsed a source corpus of 120M sentence English newspaper texts from the LCC[5] (Richter et al., 2006) with the Stanford parser (de Marneffe et al., 2006) and used dependencies to extract features for words: each dependency triple $(w1, r, w2)$ denoting a dependency of type r between words $w1$ and $w2$ results in a feature $(r, w2)$ characterizing $w1$, and a feature $(w1, r)$ characterizing $w2$[6]. After counting the frequency of each feature for each word, we apply a significance measure (log-likelihood test (LL), (Dunning, 1993)), rank features per word according to their significance, and prune the data, keeping only the 1000 most salient features (F_w) per word[7]. The similarity of two words is then given by the number of their common features. Our distributional thesaurus provides a list of the 1000 most salient features and a ranked list of up to 200 similar words (sim_w, based on the number of shared features) for all words above a certain frequency in the source corpus. We compute the following features to characterize a target word / substitution pair:

- To what extent the context c characterizes s_i: $\frac{\sum_{c\in F_{s_i}} LL(F_{s_i}(c))}{\sum_{s_j\in S}\sum_{c\in F_{s_j}} LL(F_{s_j}(c))}$
- percentage of shared words among the top k similar words to t and to s_i: $\frac{|sim_t|_k \cap |sim_{s_i}|_k}{max(|sim_t|_k, |sim_{s_i}|_k)}$, for $k = 1, 5, 10, 20, 50, 100, 200$[8]
- percentage of shared salient features among the top k features of t and s_i, globally and restricted to the words from the target sentence: $\frac{|F_t|_k \cap |F_{s_i}|_k}{max(|F_t|_k, |F_{s_i}|_k)}$ and $\frac{|F_t|_k \cap |F_{s_i}|_k \cap |c|}{|c|}$, for $k = 1, 5, 10, 20, 50, 100, 1000$
- boolean feature indicating whether $s_i \in sim_t$ or not (in top 100 similar words)

[5] http://corpora.informatik.uni-leipzig.de/

[6] open source implementation and data available at http://sourceforge.net/p/jobimtext

[7] The pruning operation greatly reduces runtime at thesaurus collection, rendering memory reduction techniques like (Charikar et al., 2004) as unnecessary.

[8] The various values for k trade off the salience of this feature for coverage: only very few substitutions have overlap in the top 1-5 similar words set, but if this happens, it is a very strong indicator of contextual fitness, whereas overlap within the top 100-200 similar words is present for much more target/substitution pairs, but it is a weaker indicator of fitness.

4.3.3 Local n-gram Features (from Web 1T)

Syntagmatic coherence, measured as the n-gram frequency of the context with the candidate substitution serves as the basis of ranking in the best Semeval 2007 system (Giuliano et al., 2007), which is also our baseline method here. We use the same n-grams as features in our supervised model:

- 1-5-gram frequencies in a sliding window around t: $freq(c_l s_i c_r)/freq(c_l t c_r)$, normalized w.r.t t

- 1-5-gram frequencies in a sliding window around t: $freq(c_l s_i c_r)/\sum freq(c_l S c_r)$, normalized w.r.t. S

- for each of x in {*'and', 'or', ','*}, 3-5-gram frequencies in a sliding window around t: $freq(c_l t x s_i c_r)/freq(c_l t c_r)$ (how frequently the target and candidate are part of a list or conjunctive phrase)

4.3.4 Shallow Syntactic Features

We also use part of speech information (from TreeTagger (Schmid, 1994)) as features, in order to enable the model to learn POS-specific patterns. This is especially important for the LexSub dataset, which contains examples from all major parts of speech (the TWSI dataset contains only noun targets). Specifically, we use:

- 1-3-grams of main POS categories in a window around t, e.g. *NVV* for a noun, verb, verb context

- Penn Treebank POS code of t

4.3.5 Example

For clarity, we exemplify our delexicalized features briefly. Using WordNet as a source for the word *bright*, we considered the 11 words *brilliant, vivid, smart, burnished, lustrous, shining, shiny, undimmed, brilliant, hopeful, promising* from the synsets of *bright*, and 64 further words from its related synsets (e.g. *intelligent, glimmery, polished, happy, ...*) as potential paraphrases. That is, for the sentence "He was **bright** and independent and proud.", where the human annotators listed *intelligent, clever* as suitable paraphrases, our system had 1 correct (*intelligent*) and 74 incorrect substituions in the candidate set (that is, *clever* is not found in WordNet in the above described way). The substitution *intelligent* in this context is characterized by a total of 178 active features. Of those, 112 features are based on local n-gram features (Sect. 4.3.3), where the large number stems from different n in n-gram, as well as the different variants of normalization and copies for the particular POS (here: JJ) and for all POS. For instance, "bright" and "intelligent" are frequently occurring in comma-separated enumerations, and "intelligent" fits well in the target context based on n-gram probabilities. The second largest block of features is constituted by 48 active distributional similarity features (Sect. 4.3.2), which are also available per POS and for different normalizations. Here is e.g. captured that the candidate has a high distributional similarity to the target with respect to our background corpus. The 12 shallow syntatic features (Sect 4.3.4) capture various present POS patterns around the target, and the 6 resource-based features (Sect. 4.3.1) e.g. inform about the number of senses of the target (10) and the candidate (4).

4.4 Results

Now, we describe our results in detail. First we compare our system on two datasets with a competitive baseline, which uses the same candidate set as our ML-based model, and the simple and effective ranking function based on Google n-grams described by Giuliano et al. (2007). Later on we analyze how the four major feature groups contribute to the results in a feature ablation experiment, and then we provide a detailed and thorough comparison to earlier works that are similar to the model presented here and used the same dataset (LexSub) for evaluation.

4.4.1 Semeval 2007 Lexical Substitution

In Table 2 we report results on the LexSub dataset. As can be seen, our model outperforms the baseline by a significant margin ($p < 0.01$ for all measures, using a paired t-test for significance). Both the overall rankings and the P@1 scores are of higher quality than the rankings based only on n-grams.

4.4.2 Turk Bootstrap Word Sense Inventory

The results on the TWSI dataset are provided in Table 3. Our model outperforms the baseline in all

	cand. from WN		from Gold St.	
	GAP	P@1	GAP	P@1
Baseline	36.8	31.1	46.9	49.5
Our model	43.8	40.2	52.4	57.7

Table 2: Comparison to the baseline on LexSub 2007.

	cand. from WN		from Gold St.	
	GAP	P@1	GAP	P@1
Baseline	33.8	28.2	44.4	44.5
Our model	36.6	32.4	47.2	49.5

Table 3: Comparison to the baseline on the TWSI dataset.

the comparisons similar to the LexSub dataset. The differences are not so pronounced but still highly significant ($p < 0.01$). This is consistent with the observation by several Semeval 2007 participants and with a per-POS analysis of our results on LexSub: the ranking task seems to be more challenging for nouns than for other parts of speech. When using WordNet, for about half (11165/22543) of the instances, individual scores are 0 (cf. Table 1). For the other half, avg. P@1 score is around 0.7, which results in 0.324 overall. Note that the task of ranking in avg. 7.5 items is considerably easier than ranking in avg. 22 items, which explains the high P@1 scores for cases where good candidates exist – also, a random ranker would score higher in this case.

These results demonstrate that the proposed delexicalized approach is superior to a competitive baseline across two datasets.

4.5 Feature Exploration

We explored the contribution of our various feature types on the LexSub dataset with candidate set from the gold standard. Our MaxEnt model relying only on local n-gram frequency features, i.e. the same information as the baseline model, achieved a GAP score of 48.3 and P@1 of 52.1, respectively. This result is significantly better than the baseline ($p < 0.01$), i.e. the machine learnt ranking model is better than a state-of-the-art handcrafted ranker based on the same data. All single feature groups, when combined with n-grams, lead to significant improvements ($p < 0.01$), which proves the usefulness of each feature group. In order to assess the contribution of each group to the *overall* system performance, we performed a feature ablation experiment. That is, we trained the MaxEnt model with using all feature groups (this equals the model in Table 2) and then with leaving each of the feature groups out once. As can be seen, all feature groups improve the overall results in a noticeable way, i.e. their contribution is complementary.

	GAP	P@1
w/o n-gram features	47.3	48.9
w/o distr. thesaurus	49.8	55.0
w/o POS features	51.6	56.3
w/o WN features	51.7	57.0
Our model (all)	**52.4**	**57.7**

Table 4: Feature ablation experiment (on LexSub dataset, with candidates from Gold Standard).

4.5.1 Comparison to Previous Works

In Table 5 we compare our method with previous works in the field, using the LexSub dataset.

candidates from WN		from Gold Standard	
	Best-P		GAP
		PadóErk10	38.6
Giuliano	12.93	DinuLapata	42.9
Martinez	12.68	Thater10	46.0
Sinha	13.60	Thater11	51.7
Baseline	11.75	Baseline	46.9
Our model	**15.94**	Our model	**52.4**

Table 5: Comparison to previous works (LexSub dataset).

In the left column of Table 5, we compare the performance of our system to representative Semeval 2007 participants, namely Martinez et al. (2007) and Giuliano et al. (2007). In order to make a fair comparison, we report scores for the official test data of Semeval 2007, using a 10-fold cross-validation scheme. Martinez et al. (2007) developed their system based on WordNet and we use the same candidate set here that they proposed in their system description. Our reimplementation of (Giuliano et al., 2007) performs below the original scores, due to the more restricted source of substitution candidates (they use more lexical resources), yet uses the same ranking methodology based on Google n-grams that we adopted here as our baseline. We also report the best previous result for this task, which was achieved via the (supervised) combination of lexical resources to improve the performance (Sinha and Mihalcea, 2009). Our model outperforms this result by a large margin for the best-precision evaluation (mode-P, precision measured on those examples where there is a clear best substitution provided by humans was 26.3%, compared to 21.3% reported by Sinha and Mihalcea (2009). This is especially promising in light of the fact that we use only a single source (WordNet) for synonyms and achieve our improvements through more advanced delexicalized features in an improved ranking model. Sinha and Mihalcea (2009), on the other hand, used comparably simple features for contextualization, of which n-gram features were deemed most successful. As Sinha and Mihalcea (2009) showed improvements through utilizing several synonym sources, a combination of their approach with ours should allow for further improvements in the future.

In the right column of Table 5, we compare our model to previous works that addressed only the ranking task, and report performance on the whole dataset (i.e. trial and test). As can be seen, the methodology proposed here outperforms previous ranking models, without the need to develop a high-quality ranking model by hand, and without the need to parse the test sentences. Our delexicalized supervised model only requires the development of features, and achieves excellent results without major task-specific tuning or customization: we omitted the optimization of the feature set and the parameters of the learning model. This fact makes us assume that the proposed model can be applied more quickly and easily than previous models that have several important design aspects to choose from.

5 Conclusion and Future Work

In this study, we presented a supervised approach to all-words lexical substitution based on delexicalized features, which enables us to fully exploit the power of supervised models while ensuring applicability to a large, open vocabulary.

Results demonstrate the feasibility of this method: our MaxEnt-based ranking approach improved over the baseline in all settings and yielded – to our knowledge – the best scores for lexical substitution with automatically gathered synonyms on the Semeval 2007 LexSub dataset. Also, it performed slightly better than the state of the art for candidates pooled from the gold standard without any parameter tuning or empirical design choices.

In this study, we established transparency between Semeval-style and ranking-only studies in lexical substitution – two lines of work that were difficult to compare in the past. Further, we observe similar improvements on two different datasets, showing the robustness of the approach.

While previous works showed the potential of more/improved lexical resources for lexical substitution, we improved over the best Semeval-style performance just by exploiting an improved ranking model over a standard WordNet-based candidate set. These results indicate that improvements from lexical resources and better ranking models are additive, thus we want to add more lexical resources in our system in the future.

Of course there are several other ways to improve further the work described here. First of all, similar to the best ranking approaches (e.g. Thater et al. (2011)), one could use contextualized feature functions to make global information from the distributional thesaurus more accurate. Instead of using globally calculated similarities, information from the distributional thesaurus could be contextualized via constraining the statistics with words from the context.

Other natural ways to improve the model described here are to make heavier use of parser information or to employ pair-wise or list-wise machine learning models (Cao et al., 2007), which are specifically designed for subset ranking. Lastly, while intrinsic evaluation of lexical substitution is important, we would like to show its practicability in tasks such as steganography or information retrieval.

Acknowledgements

This work has been supported by the Hessian research excellence program Landes-Offensive zur Entwicklung Wissenschaftlich-ökonomischer Exzellenz (LOEWE) as part of the research center *Digital Humanities*, and by the German Ministry of Education and Research under grant *SiDiM* (grant no. 01IS10054G).

References

Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. 2012. Semeval-2012 task 6: A pilot on semantic textual similarity. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 385–393, Montréal, Canada.

Chris Biemann. 2012. Creating a System for Lexical Substitutions from Scratch using Crowdsourcing. *Language Resources and Evaluation: Special Issue on Collaboratively Constructed Language Resources*, 46(2).

Róbert Busa-Fekete, Balázs Kégl, Éltető Yamás, and Györgi Szarvas. 2011. A robust ranking methodology based on diverse calibration of adaboost. In *European Conference on Machine Learning*, volume LNCS, 6911, pages 263–279.

Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. 2007. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24rd International Conference on Machine Learning*, pages 129–136.

Ching-Yun Chang and Stephen Clark. 2010. Practical linguistic steganography using contextual synonym substitution and vertex colour coding. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1194–1203, Cambridge, MA.

Moses Charikar, Kevin Chen, and Martin Farach-Colton. 2004. Finding frequent items in data streams. *Theor. Comput. Sci.*, 312(1):3–15.

D. Cossock and T. Zhang. 2008. Statistical analysis of Bayes optimal subset ranking. *IEEE Transactions on Information Theory*, 54(11):5140–5154.

Ido Dagan, Oren Glickman, Alfio Gliozzo, Efrat Marmorshtein, and Carlo Strapparava. 2006. Direct word sense matching for lexical substitution. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, ACL-44, pages 449–456, Sydney, Australia.

Marie-Catherine de Marneffe, Bill MacCartney, and Christopher D. Manning. 2006. Generating typed dependency parses from phrase structure parses. In *LREC 2006*, Genova, Italy.

Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407.

Georgiana Dinu and Mirella Lapata. 2010. Measuring distributional similarity in context. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1162–1172, Cambridge, MA.

Ted Dunning. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74.

Katrin Erk and Sebastian Padó. 2008. A structured vector space model for word meaning in context. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 897–906, Honolulu, Hawaii.

Katrin Erk and Sebastian Padó. 2010. Exemplar-based models for word meaning in context. In *Proceedings of the ACL 2010 Conference Short Papers*, pages 92–97, Uppsala, Sweden.

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

Evgeniy Gabrilovich and Shaul Markovitch. 2007. Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pages 1606–1611.

Bela Gipp, Norman Meuschke, and Joeran Beel. 2011. Comparative Evaluation of Text- and Citation-based Plagiarism Detection Approaches using GuttenPlag. In *Proceedings of 11th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL'11)*, pages 255–258, Ottawa, Canada. ACM New York, NY, USA. Available at http://sciplore.org/pub/.

Claudio Giuliano, Alfio Gliozzo, and Carlo Strapparava. 2007. FBK-irst: Lexical substitution task exploiting domain and syntagmatic coherence. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 145–148, Prague, Czech Republic.

Samer Hassan, Andras Csomai, Carmen Banea, Ravi Sinha, and Rada Mihalcea. 2007. UNT: SubFinder: Combining knowledge sources for automatic lexical substitution. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 410–413, Prague, Czech Republic.

Kazuaki Kishida. 2005. *Property of Average Precision and Its Generalization: An Examination of Evaluation Indicator for Information Retrieval Experiments*. NII technical report. National Institute of Informatics.

Ping Li, Christopher J.C. Burges, and Qiang Wu. 2007. McRank: Learning to rank using multiple classification and gradient boosting. In *Advances in Neural Information Processing Systems*, volume 19, pages 897–904. The MIT Press.

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.

Dekang Lin. 1998. Automatic retrieval and clustering of similar words. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics*, volume 2 of *ACL '98*, pages 768–774, Montreal, Quebec, Canada.

David Martinez, Su Nam Kim, and Timothy Baldwin. 2007. MELB-MKB: Lexical substitution system based on relatives in context. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 237–240, Prague, Czech Republic.

Andrew Kachites McCallum. 2002. *MALLET: A Machine Learning for Language Toolkit*. http://mallet.cs.umass.edu.

Diana McCarthy and Roberto Navigli. 2007. Semeval-2007 task 10: English lexical substitution task. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 48–53, Prague, Czech Republic.

Rada Mihalcea and Andras Csomai. 2005. Senselearner: word sense disambiguation for all words in unrestricted text. In *Proceedings of the ACL 2005 on Interactive poster and demonstration sessions*, ACLdemo '05, pages 53–56.

Joseph Reisinger and Raymond Mooney. 2010a. A mixture model with sharing for lexical semantics. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1173–1182, Cambridge, MA.

Joseph Reisinger and Raymond J. Mooney. 2010b. 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*, pages 109–117, Los Angeles, California.

M. Richter, U. Quasthoff, E. Hallsteinsdóttir, and C. Biemann. 2006. Exploiting the leipzig corpora collection. In *Proceesings of the IS-LTC 2006. Ljubljana, Slovenia.*

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

Ravi Sinha and Rada Mihalcea. 2009. Combining lexical resources for contextual synonym expansion. In *Proceedings of the International Conference RANLP-2009*, pages 404–410, Borovets, Bulgaria.

Ravi Som Sinha and Rada Flavia Mihalcea. 2011. Using centrality algorithms on directed graphs for synonym expansion. In R. Charles Murray and Philip M. McCarthy, editors, *FLAIRS Conference*. AAAI Press.

Stefan Thater, Hagen Fürstenau, and Manfred Pinkal. 2010. Contextualizing semantic representations using syntactically enriched vector models. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 948–957, Uppsala, Sweden.

Stefan Thater, Hagen Fürstenau, and Manfred Pinkal. 2011. Word meaning in context: A simple and effective vector model. In *Proceedings of the Fifth International Joint Conference on Natural Language Processing : IJCNLP 2011*, pages 1134–1143, Chiang Mai, Thailand. MP, ISSN 978-974-466-564-5.

Umut Topkara, Mercan Topkara, and Mikhail J. Atallah. 2006. The hiding virtues of ambiguity: quantifiably resilient watermarking of natural language text through synonym substitutions. In *Proceedings of the 8th workshop on Multimedia and security*, pages 164–174, New York, NY, USA. ACM.