Language Transfer Learning for Supervised Lexical Substitution

Gerold Hintz and **Chris Biemann**
Research Training Group AIPHES / FG Language Technology
Computer Science Department, Technische Universität Darmstadt
{hintz,biem}@lt.informatik.tu-darmstadt.de

Abstract

We propose a framework for lexical substitution that is able to perform transfer learning across languages. Datasets for this task are available in at least three languages (English, Italian, and German). Previous work has addressed each of these tasks in isolation. In contrast, we regard the union of three shared tasks as a combined multilingual dataset. We show that a supervised system can be trained effectively, even if training and evaluation data are from different languages. Successful transfer learning between languages suggests that the learned model is in fact independent of the underlying language. We combine state-of-the-art unsupervised features obtained from syntactic word embeddings and distributional thesauri in a supervised delexicalized ranking system. Our system improves over state of the art in the full lexical substitution task in all three languages.

1 Introduction

The lexical substitution task is defined as replacing a target word in a sentence context with a synonym, which does not alter the meaning of the utterance. Although this appears easy to humans, automatically performing such a substitution is challenging, as it implicitly addresses the problem of both determining semantically similar substitutes, as well as resolving the ambiguity of polysemous words. In fact, lexical substitution was originally conceived as an extrinsic evaluation of Word Sense Disambiguation (WSD) when first proposed by McCarthy & Navigli (2007). However, a system capable of replacing words by appropriate meaning-preserving substitutes can be utilized in downstream tasks that require paraphrasing of input text. Examples of such use cases include text simplification, text shortening, and summarization. Furthermore, lexical substitution can be regarded as an alternative to WSD in downstream tasks requiring word disambiguation. For example, it was successfully applied in Semantic Textual Similarity (Bär et al., 2012). A given list of substitution words can be regarded as a vector representation modeling the meaning of a word in context. As opposed to WSD systems, this is not reliant on a predefined sense inventory, and therefore does not have to deal with issues of coverage, or sense granularity. On the other hand, performing lexical substitution is more complex than WSD, as a system has to both generate and rank a list of substitution candidates per instance.

Over the last decade, a number of shared tasks in lexical substitution has been organized and a wide range of methods have been proposed. Although many approaches are in fact language-independent, most existing work is tailored to a single language and dataset. In this work, we investigate lexical substitution as a multilingual task, and report experimental results for English, German and Italian datasets. We consider a supervised approach to lexical substitution, which casts the task as a ranking problem (Szarvas et al., 2013b). We adapt state-of-the-art unsupervised features (Biemann and Riedl, 2013; Melamud et al., 2015a) in a delexicalized ranking framework and perform transfer learning experiments by training a ranker model from a different language. Finally, we demonstrate the utility of aggregating data from different languages and train our model on this single multilingual dataset. We are able to improve the state of the art for the full task on all datasets.

The remainder of this paper is structured as follows. In Section 2 we elaborate on the lexical substitution task and datasets. Section 3 shows related work of systems addressing each of these tasks. In Section 4 we describe our method for building a supervised system capable of transfer learning. Section 5 shows our experimental results and discussion. Finally in Section 6 we give a conclusion and outlook to future work.

2 Lexical substitution datasets and evaluation

The lexical substitution task was first defined at *SemEval 2007* (McCarthy and Navigli, 2007, "SE07"). A lexical sample of target word is selected from different word classes (nouns, verbs, and adjectives). Through annotation, a set of valid substitutes was collected for 10-20 contexts per target. Whereas in the original SE07 task, annotators were free to provide “up to three, but all equally good” substitutes, later tasks dropped this restriction. Substitutes were subsequently aggregated by annotator frequency, creating a ranking of substitutes. The use of SE07 has become a de-facto standard for system comparison, however equivalent datasets have been produced for other languages. *Evalita 2009* posed a lexical substitution task for Italian (Toral, 2009, "EL09"). Participants were free to obtain a list of substitution candidates in any way, most commonly *Italian WordNet*[1] was used. A *WeightedSense* baseline provided by the organizers proved very strong, as all systems scored below it. This baseline is obtained by aggregating differently weighted semantic relations from multiple human-created lexical resources (Ruimy et al., 2002). A German version of the lexical substitution task was organized at *GermEval 2015* (Cholakov et al., 2014; Miller et al., 2015, "GE15"). Likewise, *WeightedSense* was able to beat both of two participating systems in *oot* evaluations (Miller et al., 2015).

A variation for *cross-lingual lexical substitution* was proposed by Mihalcea et al. (2010), in which substitute words are required in a different language than the source sentence. The sentence context as well as the target word were given in English, whereas the substitute words should be provided in Spanish (annotators were fluent in both languages). This variant is motivated by direct application in Machine Translation systems, or as an aid for human-based translation. There also exists a larger crowd-sourced dataset of 1012 nouns (Biemann, 2013, "TWSI"), as well as an all-words dataset in which all words in each sentence are annotated with lexical expansions (Kremer et al., 2014). Evaluation of lexical substitution adheres to metrics defined by SE07 (McCarthy and Navigli, 2007), who provide two evaluation settings[2]; *best* evaluating only a system’s “best guess” of a single target substitute and *oot,* an unordered evaluation of up to ten substitutes. Thater et. al (2009) proposed to use *Generalized Average Precision* (GAP), to compare an output *ranking* rather than unordered sets of substitutes.

Dataset comparison The proposed lexical substitution datasets (SE07, EL09, GE15) differ in their degree of ambiguity of target items. If a dataset contains mostly target words that are unambiguous, substitution lists of different instances of the same target are similar, despite occurring in different context. We can quantify this degree of variation by measuring the overlap of gold substitutes of each target across all contexts. For this, we adapt the *pairwise agreement* (PA) metric defined by McCarthy & Navigli (2009). Instead of inter-annotator agreement we measure agreement across different context instances. Let T be a set of lexical target words, and D dataset of instances $(t_i, S_i) \in D$, in which target $t_i \in T$ is annotated with a set of substitutes S_i. Then we regard for each target word t the substitute sets $S_t \subset D$ for t. We define a *substitute agreement* as $SA(t)$ as the mean pairwise dice coefficient between all $s_1, s_2 \in S_t$ where $s_1 \neq s_2$. For each dataset D we list the substitute variance $SV = 1 - \frac{1}{|T|} \sum_{t \in T} SA(t)$. Table 1 shows this metric for the three datasets, as well as for subsets of the dataset according to target part of speech. It can be seen that the variance in gold substitutes differs substantially between datasets, but not much between target word type within a dataset. *EL09* has the highest degree of variance, suggesting that targets tend to be more ambiguous, whereas *GE15* has the lowest degree of variance, suggesting less ambiguity.

3 Related Work

Lexical substitution has been addressed extensively in recent years. Early systems, having only very few training instances available, use un-

[1] Italian WordNet has later been migrated into MultiWordNet (MWN), which is used in this work.

[2] The original SE07 task had a third evaluation setting MWE, in which systems had to correctly identify which target words were part of a multiword expression.

dataset	substitute variance (*SV*)				
	noun	verb	adj	adv	all
SemEval-2007	0.78	0.79	0.72	0.66	0.75
Evalita-2009	0.84	0.82	0.83	0.82	0.83
GermEval-2015	0.59	0.67	0.60	-	0.66
all	0.75	0.72	0.73	0.69	0.73

Table 1: Degree of variation in gold answers

supervised approaches for determining appropriate substitutes. For the English SE07 task, systems mostly consider substitution candidates from *WordNet* (Fellbaum, 1998) and cast lexical substitution into a ranking task. Experiments may also be performed by pooling the set of candidates from the gold data, evaluating a pure ranking variant. Early approaches use a contextualized word instance representation and rank candidates according to their similarity to this representation. Effective representations are *syntactic vector space models* (Erk and Padó, 2008; Thater et al., 2011), which use distributional sparse vector representations based on the syntactic context of words. Performance improvement could be shown for different models, including the use of graph centrality algorithms on directional word similarity graphs (Sinha and Mihalcea, 2011), and clustering approaches on word instance representations (Erk and Padó, 2010). Multiple systems have built upon the distributional approach. Extensions include the use of LDA topic models (Ó Séaghdha and Korhonen, 2014), and probabilistic graphical models (Moon and Erk, 2013). The current state of the art combines a distributional model with the use of n-gram language models (Melamud et al., 2015a). They define the context vector of each word in a background corpus as a *substitute vector*, which is a vector of suitable filler words for the current n-gram context. They then obtain a contextualized *paraphrase vector* by computing a weighted average of substitute vectors in the background corpus, based on their similarity to the current target instance. In contrast to traditional sparse vector representations obtained through distributional methods, a recent trend is the use of low-dimensional dense vector representations. The use of such vector representations or word embeddings has been popularized by the *continuous bag-of-words* (CBOW) and *Skip-gram* model (Mikolov et al., 2013a). Melamud et al. (2015b) show a simple and knowledge-lean model for lexical substitution based solely on syntactic word embeddings. As we leverage this model as a feature in our approach, we will elaborate on this in Section 4. Another approach for applying word embeddings to lexical substitution is their direct extension with multiple word senses, which can be weighted according to target context (Neelakantan et al., 2014).

Biemann (2013) first showed that the lexical substitution task can be solved very well when sufficient amount of training data is collected per target. An approach based on crowdsourcing human judgments achieved the best performance on the S07 dataset to day. However, judgments had to be collected for each lexical item, and as a consequence the approach can not scale to an open vocabulary. As an alternative to *per-word* supervised systems trained on target instances per lexeme, *all-words* systems aim to generalize over all lexical items. Szarvas et al. (2013a) proposed such a system by using *delexicalization*: features are generalized in such a way that they are independent of lexical items, and thus generalize beyond the training set and across targets. Originally, a maximum entropy classifier was trained on target-substitute instances and used for pointwise ranking of substitution candidates. In a follow-up work it was shown that learning-to-rank methods could drastically improve this approach, achieving state-of-the-art performance with a *LambdaMART* ranker (Szarvas et al., 2013b). In this work we will build upon this model and further generalize not only across lexical items but across different languages.

For both EL09 and GE15, existing approaches have been adapted. For the Italian dataset, a distributional method was combined with LSA (De Cao and Basili, 2009). The best performing system applied a WSD system and language models (Basile and Semeraro, 2009). For the German dataset, Hintz and Biemann (2015) adapted the supervised approach by (Szarvas et al., 2013a), achieving best performance for nouns and adjectives. Jackov (2015) used a deep semantic analysis framework employing an internal dependency relation knowledge base, which achieved the best performance for verbs.

4 Method description

We subdivide lexical substitution into two subtasks; *candidate selection* and *ranking*. For a given target t, we consider a list of possible substitutes $s \in C_t$, where C_t is a static per-target candidate list. Our method is agnostic to the creation of this static resource, which can be obtained either by an unsupervised similarity-based approach, or from a lexical resource. In particular, candidates obtained at this stage do not disambiguate possible multiple senses of t, and are filtered and ordered in the ranking stage by a supervised model.

In modeling a supervised system, we have experimented with two learning setups. The first is applying a standard classification / regression learner. Here, lexical substitution is cast into a pointwise ranking task by training on target-substitute pairs generated from the gold standard. For each sentence context c, target word t and substitute s, we regard the tuple (c,t,s) as a training instance. We obtain these training instances for each lexsub instance (c,t) by considering all substitutes $s \in G_t \cup C_t$ where G_t are all candidates for target t pooled from the gold data and C_t are obtained from lexical resources. We then experiment with two labeling alternatives for a binary classification and a regression setup, respectively. For binary classification we label each instance (c,t,s) as *positive* if s has been suggested as a substitute for t by at least one annotator, and as *negative* otherwise. For regression, we normalize the annotation counts for each substitute to obtain a score in $(0,1]$ if a substitute s occurs in the gold data, 0 otherwise. The ranking of substitutes per target is obtained by considering the posterior likelihood of the *positive* label as yielded by a classifier model. We have tried multiple classifiers but have found no significant improvement over a *maximum entropy* baseline[3]. Our second setup is a learning-to-rank framework, adapted from (Szarvas et al., 2013b). Here, we are not restricted to a pointwise ranking model, but consider pairwise and listwise models[4].

We base our feature model on existing research. In addition to basic syntactic and frequency-based features, we obtained sophisticated features from trigram and syntactic thesauri, motivated by the findings of Biemann and Riedl (2013), as well as syntactic embedding features motivated by Melamud et al. (2015b).

dataset	maximum recall	
	w/ MWE	w/o MWE
SemEval-2007	0.459	0.404
Evalita-2009	0.369	0.337
GermEval-2015	0.192	0.178
all	0.242	0.223

Table 2: Upper bound for substitute recall based on lexical resources *WordNet*, *MultiWordNet*, *GermaNet*

4.1 Candidate selection

We confirm earlier research (Sinha and Mihalcea, 2009) on the high quality of selecting candidates from lexical resources. We thus base our candidate selection on prevalently used resources: *WordNet* (Fellbaum, 1998) for English, *GermaNet* (Hamp and Feldweg, 1997) for German and *MultiWordNet* (Pianta et al., 2002) for Italian. For all resources, we consider all possible senses for a given target word and obtain all *synonyms*, *hypernyms* and *hyponyms* and their transitive hull. Thus, for the *hypernymy* and *hyponymy* relation, we follow the respective edges in the graph collecting all nodes (synsets) along the path. For each synset, we extract all lemmas as substitution candidates. Although restricting candidates incurs a relatively low upper bound on system recall, we still obtain best results using this rather conservative filter. Table 2 shows the upper bound for system recall for each of the datasets, evaluated with and without removing all multiword expressions from both candidate lists and gold data. A higher coverage of *WordNet* is a plausible explanation for the much higher recall on the English data.

4.2 Supervised ranking

Learning-to-rank methods train a supervised model for ranking a list of items by *relevance*. A basic *pointwise* approach applies regression techniques to obtain a relevance scores for each item in isolation. More advanced models are based on *pairwise* preference information for instance pairs, and *listwise* approaches, which are optimized on a global metric of a given ranking output. An extensive overview of learning-to-rank models can be found in (Burges, 2010). For lexical substitution, *LambdaMART* (Wu et al., 2010) has been found to be particularly effective. *LambdaMART* is a listwise method based on gradient boosting of regression trees. Its two main hyperparameters are

[3] For classification setup we use *Mallet*: `http://mallet.cs.umass.edu/`

[4] For learning-to-rank we use *RankLib*: `http://mallet.cs.umass.edu/`

the number of leaves in each regression tree and the number of iterations and trees. We have not performed extensive tuning of these hyperparameters and used default settings, an ensemble of 1000 trees with 10 leaves.

4.3 Delexicalized features

The idea of delexicalization has been proposed, for instance, by Bergsma et al. (2007). They propose to use statistical measures based solely on the frequency of different expansions of the same target term. Their feature set has motivated a large subset of the feature model, which we adapt in this work. The idea of generalizing features for lexical substitution in such a way that they work across lexical items has been shown by Moon and Erk (2013), and made explicit by Szarvas et al. (2013a). Instances are characterized using non-lexical features from heterogeneous evidence. The intuition of this feature model is to exploit redundant signals of substitutability from different sources and methods.

In cases where background corpora are required, the following data is used throughout all features: For English, a newspaper corpus compiled from 105 million sentences from the Leipzig Corpora Collection (Richter et al., 2006) and the Gigaword corpus (Parker et al., 2011) was used. For German a 70M sentence newswire corpus (Biemann et al., 2007) was used. For Italian, a subset of 40M sentences of *itWac*, a large webcrawl, was used (Baroni et al., 2009).

Shallow syntactic features We apply a part-of-speech tagger trained on universal POS tags (Petrov et al., 2012), which we simplify into the classes *noun*, *verb*, *adjective* and *adverb*. Using these simplified tags we construct an *n*-gram sliding window, with $n \in [1..5]$, of POS around the target. We could also reduce window sizes drastically to $n = 1,2$ without sacrificing performance.

Frequency features We use language models for each of the languages to obtain *frequency ratio* features. An *n*-gram sliding window around a target t is used to generate a set of features $\frac{freq(c_l,s,c_r)}{freq(c_l,t,c_r)}$, where c_l and c_r are the left and right context words around t. Here, we normalize the frequency of the substitute with the frequency of the n-gram with original target t. As a variant, we further normalize frequencies by the set of all substitutes, to obtain frequencies features $\frac{freq(c_l,s,c_r)}{\sum_{s' \in C_t} freq(c_l,s',c_r)}$ where C_t is the set of candidate substitutes for t. In our experiments we used sliding windows of size $[1..5]$. We obtain 5-gram counts from *web1t* (Brants and Franz, 2009).

Conjunction ratio features Based on the n-gram resources above, we further define a conjunctive phrase ratio feature, which measures how often the construct $(c_l, t, conjunction, s, c_r)$ occurs in a background corpus; i.e. how often t and s co-occur with a *conjunction* word ("*and*", "*or*", ","), within the context of the sentence. As there is a different set of conjunction words for each language, we first aggregate the mean over all conjunction words:

$$conj_{l,r}(t,s) = \frac{1}{|CONJ|} \sum_{con \in CONJ} freq(c_l, t, con, s, c_r)$$

where l and r is the size of the left and right context window, and $CONJ$ is a set of conjunction words per-language[5]. For left and right context size $l = r = 0$ this feature also captures a context-independent conjunction co-occurrence between only t and s. Again, we normalize this feature over the set of all candidates:

$$\frac{conj_{l,r}(t,s)}{\sum_{s' \in C_t} conj_{l,r}(t,s)}$$

Distributional features We construct a *distributional thesaurus* (DT) for each of the languages by following Biemann and Riedl (2013) and obtain first-order word-to-context measures, as well as second-order word-to-word similarity measures. As context features we have experimented with both syntactic dependencies as well as left and right neighboring words, and have found them to perform equivalently. As a salience measure we use *Lexicographer's Mutual Information* (Bordag, 2008) and prune the data, keeping only the 1000 most salient features per word. Word similarity is obtained from an overlap count in the pruned context features. We model features for the *contextualized* distributional similarity between t and s as

- percentage of shared context features for the top-k context features of t and s, globally and restricted to sentence context (k =5, 20, 50, 100, 200)

[5] Conjunction words used are *and*, *or*, (*comma*), for English; *und*, *oder*, (*comma*) for German and *e*, *ed*, *o*, *od*, (*comma*) for Italian.

- percentage of shared words for the top-k similar words of t and s (k =200)
- sum of salience score of context features of s overlapping with the sentence context
- binary occurrence of s in top-k similar words of t (k =100, 200)

With the exception of the last feature, these measures are scaled to $[0,1]$ over the set of all substitute candidates.

Syntactic word embeddings We adapt the unsupervised approach by (Melamud et al., 2015a) as a set of features. We follow (Levy and Goldberg, 2014) to construct dependency-based word embeddings; we obtain syntactic contexts by running a syntactic dependency parser[6], and computing word embeddings using dependency edges as context features[7]. The resulting dense vector representations for words and context live within the same vector space. We compute the semantic similarity between a target and a substitute word from the cosine similarity in the word embedding space, as well as the first-order target-to-context similarity. For a given target word t and substitute s, let C_t be the syntactic context of t and $c \in C_t$ a single context – i.e. a dependency edge attached to t; let v_t, v_s be the vector representations of t and s in the word embedding space, and v_c the vector representation of c in the context embedding space. Then $Sim_1 = \cos(v_s, v_c)$ and $Sim_2 = \cos(v_s, v_t)$ are the first-order and second-order substitutability measures considered by Melamud et al. (2015a). In contrast to their approach, we do not just consider an unsupervised combination of these two measures, but instead use both Sim_1 and Sim_2 as single features. We also use their combinations of a *balanced* / *unbalanced*, *arithmetic* / *geometrical* mean, to obtain six numeric features in total. Importantly, these features are independent of the underlying embedding vectors and can therefore generalize across arbitrary embeddings between languages.

Semantic resource features To generalize across multiple languages we minimize the complexity of features obtained from semantic resources – which may differ notably in size and structure. From the resources listed in Section 4.1 we extract binary features for the semantic relations *synonymy*, *hypernymy* and *hyponymy*, occurring between t and s. We have also experimented with graded variants for transitive relations, such as encoding n-th level hypernymy, but have not observed any gain from this feature variation.

4.4 Transfer learning

Transfer learning is made feasible by a fully lexeme-independent and language-independent feature space. Language-specific knowledge resides only within the respective resources for each language, and gets abstracted in feature extraction. Figure 1 illustrates this process at the example of two entirely unrelated sentences in different languages (English and German). A further mediator for transfer learning is a model based on boosted decision trees. As opposed to linear models, which could not be reasonably learned across languages, a *LambdaMART* ranker is able to learn feature interaction across languages. To give an example of what the resulting model can pick up on, we can regard conditionally strong features. Consider the n-gram pair frequency ratio feature of window size $(l,r) = (1,0)$, which compares the frequency ratio of the target and substitute including a single left context word. Depending on the POS window, this feature can be highly informative in some cases, where it is less informative in others. For *adjective-noun* pairs, in which the noun is the substitution target, the model can learn that this frequency ratio is strongly positively correlated; in this case, the substitute frequently occurs with the same adjective than the original target. For other POS windows, for example *determiner-noun* pairs, the same frequency ratio may be less indicative, as most nouns frequently occur with a determiner. This property works across languages, as long as as attributive adjectives are prepositioned. In our subset of languages, this is the case for English and German, but not for Italian, which uses postpositive adjectives. Nevertheless, we are able to learn such universal feature interactions.

5 Results and discussion

Evaluation of lexical substitution requires special care, as different evaluation settings are used

[6] We trained models for *Mate* (`https://code.google.com/p/mate-tools/`) based on *universal dependencies* (`http://universaldependencies.org/`)

[7] We used *word2vecf* (`https://bitbucket.org/yoavgo/word2vecf`) for computing syntactic word embeddings

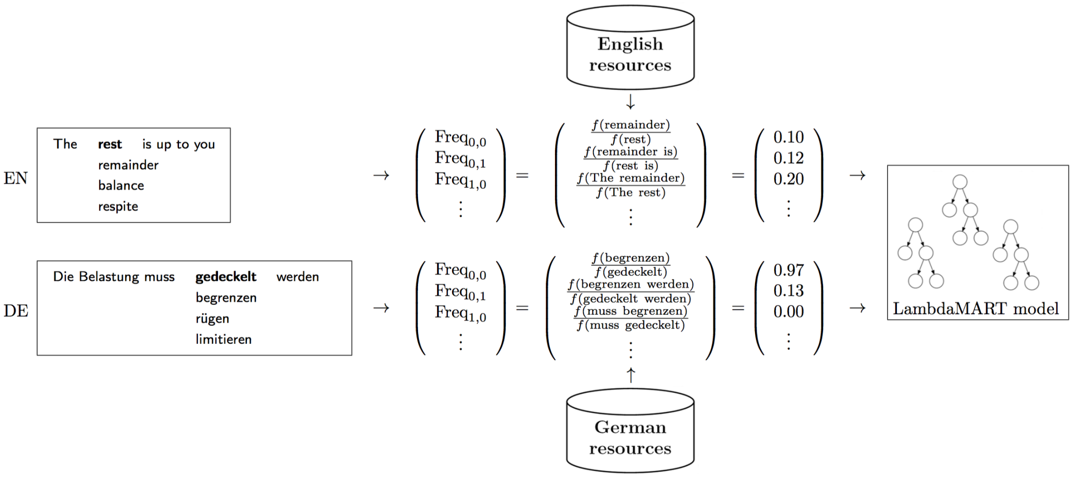

Figure 1: Visualization of feature extraction and delexicalization. Two unrelated sentences in English and German (translation: "*the strain has to be **limited***") are shown. Language-specific knowledge is obtained from resources for each language respectively. The resulting feature space is delexicalized and language independent.

throughout previous work and comparability is not always guaranteed. We follow the convention of reporting the full lexical substitution task (both generating and ranking candidates) with the metrics *P-best* and *P-oot* and report the ranking-only task (candidates pooled from the gold standard) with the *GAP* score. We further observe that previous work commonly discards multiword expressions from both the candidate lists as well as the gold data[8]. We follow this convention, but note that our system is in fact capable of successfully ranking multiword expansions out of the box. System performance slightly decreases when including MWE, as there is virtually no overlap between those provided by the system and those in the gold standard.

For ranking we experiment with different pointwise classifiers as provided by *Mallet* (MaxEnt classification and regression) as well learning-to-rank models provided by *RankLib* (RankBoost, RankNet, LambdaMART). In line with findings in (Szarvas et al., 2013b), we observe that learning-to-rank approaches work better than a pointwise classification / regression setup throughout all languages and feature subsets. Among different rankers, we confirm LambdaMART to yield the best performance, and will only report numbers using this model. As optimization metric we have explored both *NDCG@10* and *MAP*. The *NDCG* metric can incorporate different scoring weights based on annotator overlap, however MAP directly correlates with evaluation score. We have found optimizing on MAP to yield slightly better results, even if this disregards the relative score weights between gold substitutes. For the ranking-only task, we also extended the pooled training data with additional negative examples (i.e. adding all candidates as for the full task) but observed a minor decrease in system performance.

Open evaluation (best-P / oot-P)

Training	English		German		Italian	
English	16.63	48.16	7.43	26.79	8.57	31.94
German	13.20	44.61	11.97	38.45	7.05	28.75
Italian	13.91	39.72	4.25	22.66	15.19	40.37
others	17.19	46.79	8.15	27.33	10.04	30.82
all	**17.23**	**48.83**	**12.94**	**41.32**	**16.15**	**41.29**
SOA[9]	15.94	36.37	11.20	20.14	10.86	41.46

Table 3: Transfer learning results for the open candidate task (candidates from lexical resources)

Ranking evaluation (GAP)

Training	English	German	Italian
English	51.0	26.9	44.5
German	44.3	**56.2**	42.9
Italian	36.7	22.2	48.0
others	43.7	26.7	43.9
all	**51.9**	51.3	**50.0**

Table 4: Transfer learning results on the ranking-only task (candidates pooled from gold)

[8] The omission of MWE by multiple authors has been confirmed by the authors of (Melamud et al., 2015a).

We report transfer learning results across all three datasets. Table 3 shows a transfer-learning matrix for the full lexical substitution task, whereas Table 4 shows results for the ranking-only task. For evaluation, we consistently use the complete datasets, which are roughly equal in size for all languages (~ 2000 instances). For the identity entries in this matrix, as well as training on the complete dataset ("all") we follow previous supervised work and perform 10-fold cross-validation. Splits are based on the target lexeme, so that no two instances for the same target word are in different sets. Tables 3 and 4 suggest the feasibility of transfer learning. Although models trained on the original language (identity entries of the matrix) perform best, training on a different language still yields reasonable results. Training only on a single other language, not surprisingly, yields worse results for each dataset, however combining the data from the two remaining languages ("others") can mitigate this issue to some degree. Importantly, adding the data from two additional languages consistently improves system performance throughout all datasets for the open candidate task (Table 3). It is interesting to note that in case of SE07, training on only other languages performs surprisingly well for the *best-P* score, beating even a model trained on English. A possible explanation for this is that the SE07 dataset appears to be somewhere in the middle between EL09 and GE15 in terms of substitute variance. For the ranking-only task, transfer learning seems to work a little less effectively. In case of German, adding foreign language data in fact hurts *GAP* performance. This potentially originates from a much smaller set of training instances and inconsistency of the amount and overlap of pooled candidates across different tasks (as described in Table 1). We also observe that a learning-to-rank model is essential for performing transfer learning. In case of LambdaMART, an ensemble of decision trees is constructed, which is well suited to exploit redundant signals across multiple features. Linear models resulted in worse performance for transfer learning, as the resulting weights seem to be language-specific.

Feature ablation experiments are performed for various feature groups in the full and ranking-only task (Table 5). The ablation groups correspond to the feature categories defined in Section 4.3. The *frequency* group includes plain frequency features as well as conjunction ratio features. We consider only our universal model trained on all language data (with 10-fold CV for each dataset). In case of English, the full system performs best and all feature groups improve overall performance. For other languages these results are mixed. In case of the German data, embedding features and semantic relation features seem to work well on their own, so that results for other ablation groups are slightly better. For ranking-only, embedding features seem to be largely subsumed by the combination of the other groups. Ablation of embeddings differs vastly between the full and ranking-only task; they seem to more more crucial for the full task. For all languages, semantic relations are the best feature in the full task, acting as a strong filter for candidates; in ranking-only they are more dispensable.

In summary, we observe that delexicalized transfer learning for lexical substitution is possible. Existing supervised approaches can be extended to generalize across multiple languages without much effort. Training a supervised system on different language data emphasizes that the learned model is sufficiently generic to be language independent. Our feature space constructed from heterogeneous evidence consists of many features that perform relatively weakly on their own. The resulting ranking model captures redundancy between these signals. Finally, Table 6 shows our results in comparison to previous work. Note that we omit some participating systems from the original SE07 task. The reason we did not list IRST2 (Giuliano et al., 2007) is that for out-of-ten results, the system outputs the same substitute multiple times and the evaluation scheme gives credit for each copy of the substitute. Our (and other) systems do not tamper with the metric in this way, and only yield a set of substitutes. UNT (Hassan et al., 2007) uses a much richer set of knowledge bases, not all of them easily available, to achieve slightly better *oot* scores. From our experiments, we list both a model trained per language, as well as a universal model trained on all data. The latter beats nearly all other approaches on the full lexical substitution task, despite not being optimized for a single language. Although omission of MWEs is common practice for SE07, it is unclear if this was

[9] State of the art baseline, according to previous reported results, c.f. Table 6

	English		German		Italian	
	best-P	GAP	best-P	GAP	best-P	GAP
w/o syntax	15.35	49.5	12.33	42.1	15.70	50.3
w/o frequency	17.04	48.6	13.30	54.6	15.78	51.5
w/o DT	16.88	48.8	12.18	54.6	17.65	51.8
w/o sem. relation	11.51	49.9	6.82	33.9	8.06	49.7
w/o embedding	10.05	51.5	11.51	47.1	7.17	54.4
full system	17.23	51.9	12.94	51.3	16.15	50.0

Table 5: Feature ablation results for the full and ranking-only task (universal model trained on all data)

SemEval '07			
method	**best-P**	**oot-P**	**GAP**
(Erk and Padó, 2010)	-	-	38.6
(Thater et al., 2011)	-	-	51.7
(Szarvas et al., 2013a)	15.94	-	52.4
(Szarvas et al., 2013b)	-	-	55.0
(Melamud et al., 2015b)	08.09	27.65	52.9
(Melamud et al., 2015a)	12.72	36.37	**55.2**
our method (English only)	16.63	48.16	51.0
our method (all data)	**17.23**	**48.83**	51.9
Evalita '09			
method	**best-P**	**oot-P**	**GAP**
(Basile and Semeraro, 2009)	08.16	**41.46**	-
(Toral, 2009)[12]	10.86	27.52	-
our method (Italian only)	15.19	40.37	48.0
our method (all data)	**16.15**	31.18	**50.0**
GermEval '15			
method	**best-P**	**oot-P**	**GAP**
(Hintz and Biemann, 2015)	11.20	19.49	-
(Jackov, 2015)	06.73	20.14	-
our method (German only)	11.97	38.45	56.2
our method (all data)	**12.94**	**41.32**	51.3

Table 6: Experimental results of our method compared to related work for all three lexical substitution tasks

done for EL09 and GE15. However, re-inclusion of MWE does not drastically alter results[10]. In the ranking-only variant, we are not able to beat the learning-to-rank approach by Szarvas et. al (2013b), we note however that they have performed extensive hyperparameter optimization of their ranker, which we have omitted. We are also not able to achieve *GAP* scores reported by Melamud at al. (2015b). Although we used their exact embeddings, we could not reproduce their results[11].

6 Conclusion

We are the first to model lexical substitution as a language-independent task by considering not just a single-language dataset, but by merging data from distinct tasks in English, German and Italian. We have shown that a supervised, delexicalized approach can successfully learn a single model across languages – and thus perform transfer learning for lexical substitution. We observe that a listwise ranker model such as LambdaMART facilitates this transfer learning. We have further shown that incorporating more data helps training a more robust model and can consistently improve system performance by adding foreign language training data. We extended an existing supervised learning-to-rank approach for lexical substitution (Szarvas et al., 2013b) with state-of-the-art embedding features (Melamud et al., 2015b). In our experiments, a single model trained on all data performed best on each language. In all three datasets we were able to improve the current state of the art for the full lexical substitution task. The resulting model can be regarded as language-independent; given an unannotated background corpus for computing language-specific resources and a source of substitution candidates, the system can be used almost out of the box. For obtaining substitution candidates, we still rely on lexical resources such as *WordNet*, which have to be available for each language. As future work we aim to make our approach completely knowledge-free by eliminating this dependency. We can consider substitution candidates based on their distributional similarity. First experiments confirm that this already yields a much better coverage, i.e. upper bound on recall, while introducing more noise. The remaining key challenge is to better characterize possible substitutes from bad substitutes in ranked lists of distributionally similar words, which frequently contain antonyms and co-hyponyms. We will explore unsupervised acquisition of relational similarity (Mikolov et al., 2013b) for this task.

[10] For comparison, our scores including MWE for the "all data" model are as follows (*best-P*, *oot-P*, *GAP*). EL09: 15.12, 33.92, 45.8; GE15: 12.20, 41.15, 50.0

[11] Our evaluation of (Melamud et al., 2015b), *balAdd* yields a *GAP* score of 48.8, which is likely related to different evaluation settings.

[12] baseline by task organizer

Acknowledgments

This work has been supported by the German Research Foundation as part of the Research Training Group "Adaptive Preparation of Information from Heterogeneous Sources" (AIPHES) under grant No. GRK 1994/1 and the SEMSCH project, grant No. BI 1544.

References

Daniel Bär, Chris Biemann, Iryna Gurevych, and Torsten Zesch. 2012. UKP: Computing semantic textual similarity by combining multiple content similarity measures. In *Proceedings of 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*, pages 435–440, Montreal, Canada.

Marco Baroni, Silvia Bernardini, Adriano Ferraresi, and Eros Zanchetta. 2009. The WaCky wide web: a collection of very large linguistically processed web-crawled corpora. *Language resources and evaluation*, 43(3):209–226.

Pierpaolo Basile and Giovanni Semeraro. 2009. UNIBA @ EVALITA 2009 lexical substitution task. In *Proceedings of EVALITA workshop, 11th Congress of Italian Association for Artificial Intelligence*, Reggio Emilia, Italy.

Shane Bergsma and Qin Iris Wang. 2007. Learning noun phrase query segmentation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, volume 7, pages 819–826, Prague, Czech Republic.

Chris Biemann and Martin Riedl. 2013. Text: Now in 2D! A framework for lexical expansion with contextual similarity. *Journal of Language Modelling*, 1(1):55–95.

Chris Biemann, Gerhard Heyer, Uwe Quasthoff, and Matthias Richter. 2007. The Leipzig corpora collection: monolingual corpora of standard size. In *Proceedings of Corpus Linguistics*, Birmingham, UK.

Chris Biemann. 2013. Creating a system for lexical substitutions from scratch using crowdsourcing. *Language Resources and Evaluation*, 47(1):97–122.

Stefan Bordag. 2008. A Comparison of Co-occurrence and Similarity Measures As Simulations of Context. In *Proceedings of the 9th International Conference on Computational Linguistics and Intelligent Text Processing*, CICLing 2008, pages 52–63, Haifa, Israel.

Thorsten Brants and Alex Franz. 2009. Web 1T 5-gram, 10 European languages version 1. *Linguistic Data Consortium*.

Christopher J.C. Burges. 2010. From RankNet to LambdaRank to LambdaMART: An overview. Technical Report MSR-TR-2010-82, Microsoft Research.

Kostadin Cholakov, Chris Biemann, Judith Eckle-Kohler, and Iryna Gurevych. 2014. Lexical substitution dataset for German. In *Proceedings of the 9th International Conference on Language Resources and Evaluation*, pages 1406–1411, Reykjavik, Iceland.

Diego De Cao and Roberto Basili. 2009. Combining distributional and paradigmatic information in a lexical substitution task. In *Proceedings of EVALITA workshop, 11th Congress of Italian Association for Artificial Intelligence*, Reggio Emilia, Italy.

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

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*. Wiley Online Library.

Claudio Giuliano, Alfio Gliozzo, and Carlo Strapparava. 2007. FBK-irst: Lexical substitution task exploiting domain and syntagmatic coherence. In *Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007)*, pages 145–148, Prague, Czech Republic.

Birgit Hamp and Helmut Feldweg. 1997. GermaNet - a lexical-semantic net for German. In *In Proceedings of ACL workshop Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications*, pages 9–15, Madrid, Spain.

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 4th International Workshop on Semantic Evaluations (SemEval-2007)*, pages 410–413, Prague, Czech Republic.

Gerold Hintz and Chris Biemann. 2015. Delexicalized supervised German lexical substitution. In *Proceedings of GermEval 2015: LexSub*, pages 11–16, Essen, Germany.

Luchezar Jackov. 2015. Lexical substitution using deep syntactic and semantic analysis. In *Proceedings of GermEval 2015: LexSub*, pages 17–20, Essen, Germany.

Gerhard Kremer, Katrin Erk, Sebastian Padó, and Stefan Thater. 2014. What substitutes tell us - analysis of an "all-words" lexical substitution corpus. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 540–549, Gothenburg, Sweden.

Omer Levy and Yoav Goldberg. 2014. Dependency-based word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, volume 2, pages 302–308, Baltimore, MD, USA.

Diana McCarthy and Roberto Navigli. 2007. Semeval-2007 task 10: English lexical substitution task. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 48–53, Prague, Czech Republic.

Diana McCarthy and Roberto Navigli. 2009. The English lexical substitution task. *Language resources and evaluation*, 43(2):139–159.

Oren Melamud, Ido Dagan, and Jacob Goldberger. 2015a. Modeling word meaning in context with substitute vectors. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 472–482, Denver, CO, USA.

Oren Melamud, Omer Levy, and Ido Dagan. 2015b. A simple word embedding model for lexical substitution. *VSM Workshop*. Denver, CO, USA.

Rada Mihalcea, Ravi Sinha, and Diana McCarthy. 2010. Semeval-2010 task 2: Cross-lingual lexical substitution. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 9–14, Uppsala, Sweden.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013a. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, Lake Tahoe, NV, USA.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013b. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751, Atlanta, GA, USA.

Tristan Miller, Darina Benikova, and Sallam Abualhaija. 2015. GermEval 2015: LexSub – A shared task for German-language lexical substitution. In *Proceedings of GermEval 2015: LexSub*, pages 1–9, Essen, Germany.

Taesun Moon and Katrin Erk. 2013. An inference-based model of word meaning in context as a paraphrase distribution. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 4(3):42.

Arvind Neelakantan, Jeevan Shankar, Alexandre Passos, and Andrew McCallum. 2014. Efficient non-parametric estimation of multiple embeddings per word in vector space. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1059–1069, Doha, Qatar.

Diarmuid Ó Séaghdha and Anna Korhonen. 2014. Probabilistic distributional semantics with latent variable models. *Computational Linguistics*, 40(3):587–631.

Robert Parker, David Graff, Junbo Kong, Ke Chen, and Kazuaki Maeda. 2011. English Gigaword Fifth Edition. Technical report, Linguistic Data Consortium, Philadelphia, PA, USA.

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2012. A universal part-of-speech tagset. In *Proceedings of the Eight International Conference on Language Resources and Evaluation*, pages 2089–2096, Istanbul, Turkey.

Emanuele Pianta, Luisa Bentivogli, and Christian Girardi. 2002. MultiWordNet: developing an aligned multilingual database. In *Proceedings of the 1st International WordNet Conference*, pages 293–302, Mysore, India.

Matthias Richter, Uwe Quasthoff, Erla Hallsteinsdóttir, and Chris Biemann. 2006. Exploiting the Leipzig Corpora Collection. In *Proceedings of the Information Society Language Technologies Conference 2006*, pages 68–73. Ljubljana, Slovenia.

Nilda Ruimy, Monica Monachini, Raffaella Distante, Elisabetta Guazzini, Stefano Molino, Marisa Ulivieri, Nicoletta Calzolari, and Antonio Zampolli. 2002. CLIPS, a multi-level Italian computational lexicon: a glimpse to data. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation*, pages 792–799, Las Palmas, Spain.

Ravi Sinha and Rada Mihalcea. 2009. Combining lexical resources for contextual synonym expansion. In *Proceedings of the Conference in Recent Advances in Natural Language Processing*, pages 404–410, Borovets, Bulgaria.

Ravi Som Sinha and Rada Flavia Mihalcea. 2011. Using centrality algorithms on directed graphs for synonym expansion. In *FLAIRS Conference*, pages 311–316, Palm Beach, FL, USA.

György Szarvas, Chris Biemann, Iryna Gurevych, et al. 2013a. Supervised all-words lexical substitution using delexicalized features. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1131–1141, Atlanta, GA, USA.

György Szarvas, Róbert Busa-Fekete, and Eyke Hüllermeier. 2013b. Learning to rank lexical substitutions. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1926–1932, Seattle, WA, USA.

Stefan Thater, Georgiana Dinu, and Manfred Pinkal. 2009. Ranking paraphrases in context. In *Proceedings of the 2009 Workshop on Applied Textual Inference*, pages 44–47, Singapore, Republic of Singapore.

Stefan Thater, Hagen Fürstenau, and Manfred Pinkal. 2011. Word meaning in context: A simple and effective vector model. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1134–1143, Edinburgh, UK.

Antonio Toral. 2009. The lexical substitution task at EVALITA 2009. In *Proceedings of EVALITA Workshop, 11th Congress of Italian Association for Artificial Intelligence*, Reggio Emilia, Italy.

Qiang Wu, Christopher JC Burges, Krysta M Svore, and Jianfeng Gao. 2010. Adapting boosting for information retrieval measures. *Information Retrieval*, 13(3):254–270.