Combining heterogeneous knowledge resources for improved distributional semantic models

György Szarvas*, Torsten Zesch, and Iryna Gurevych

Ubiquitous Knowledge Processing Lab, Computer Science Department,
Technische Universität Darmstadt,
Hochschulstr. 10, D-64289 Darmstadt, Germany
http://www.ukp.tu-darmstadt.de

Abstract. The Explicit Semantic Analysis (ESA) model based on term cooccurrences in Wikipedia has been regarded as state-of-the-art semantic relatedness measure in the recent years. We provide an analysis of the important parameters of ESA using datasets in five different languages. Additionally, we propose the use of ESA with multiple lexical semantic resources thus exploiting multiple evidence of term cooccurrence to improve over the Wikipedia-based measure. Exploiting the improved robustness and coverage of the proposed combination, we report improved performance over single resources in word semantic relatedness, solving word choice problems, classification of semantic relations between nominals, and text similarity.

1 Introduction

Semantic relatedness (SR) aims at measuring how related the meaning, i.e. the semantic content of two words is. Computing the SR of words finds applications in many classical Natural Language Processing (NLP) problems like Word Sense Disambiguation [24], Information Retrieval [29, 22], Cross-Language Information Retrieval [5], Text Categorization [10], Information Extraction [26], Coreference Resolution [27], or Spelling Error Detection [3].

Most of the SR measures proposed in the past have two limitations. First, they only exploit the implicit knowledge encoded in *a single* structured knowledge source like WordNet or Wikipedia, or a large text collection like the World Wide Web, but do not exploit the complementary knowledge in multiple resources through combination. Second, most measures are designed to compute relatedness between words, not between longer text segments. However, SR has important applications both on the word level (Word Sense Disambiguation, Spelling Error Correction), and on the text level (Information Retrieval, Text Categorization). Therefore, in this paper, we address the combination of the knowledge encoded in heterogeneous, independent knowledge resources to obtain better and more robust performance paying attention to direct applicability to word pairs and pairs of texts alike.

* On leave from Research Group on Artificial Intelligence of the Hungarian Academy of Sciences

For this purpose, we focus on distributional semantic relatedness, and in particular, following Gabrilovich and Markovitch [9], we employ concept vector based measures to incorporate knowledge from heterogeneous resources (representing encyclopedic knowledge and lexical information in our case) to overcome the weaknesses of single resources. In Section 3, we give a detailed overview on concept vector based measures. We also propose a new formulation of the concept vector based measure that has one less degree of freedom, i.e. it does not require any pruning of word concept vectors. In Section 4, we introduce our combined measure. We implement the combination of independent knowledge sources through combining the relatedness scores provided by concept vector measures based on single resources. As concept vector measures are applicable on the word as well as on the text level, the combined measure preserves the direct applicability to longer texts. In Section 5, we demonstrate the usefulness of the proposed approach through the successful application of the combined measure to three different NLP tasks: i) solving word choice problems, ii) classification of semantic relations between nominals, and iii) text similarity computation. We also show that the combined measure yields stable performance on the word level for different parts of speech, which was not experimentally demonstrated by previous works.

2 Related Work

In the last decade, many different approaches have been proposed to measure the semantic relatedness of natural language units (i.e. words, phrases or texts). **Structural Methods** exploit the structural information through measuring path length [3, 27, 30], computing PageRank vectors [32], or comparing link vectors [20] in a lexical semantic knowledge resource like WordNet, Wikipedia or ConceptNet. **Distributional Methods** employ distributional relatedness to compare cooccurrence patterns measuring hit counts [4, 2], comparing distributional profiles [23, 1] or concept vectors [9, 34, 13] in an underlying collection of representative texts like Wikipedia or the Web.

Structural methods are defined to compute relatedness on the word level[1]. Web-based distributional methods also operate on the word level, calculating hit count based association measures like mutual information between terms. Such methods can be extended to model text-level similarity through measuring the relatedness between all word pairs in the documents to be compared, and then aggregating the word level similarities [21]. This requires $n \cdot m$ calculations to compare two texts of sizes n and m, which can be computationally demanding or even infeasible, e.g. for web-based measures where this entails $n \cdot m$ queries to a search engine.

In contrast to structural and web-based approaches operating solely on the word level, concept vector based methods using a closed collection have the advantage that longer texts can be represented similar to single words. Thus,

[1] Words are the natural unit of representation in the underlying structured resource, e.g. lexical units for WordNet and concepts (i.e. article names) for Wikipedia.

it is straightforward to compute the similarity of longer text segments [6, 9]. Concept vector based measures are applicable both on the word and text level using exactly the same formulation, and comparison of longer texts does not require substantial extra computation (i.e. direct comparison of all word pairs).

The performance of concept vector based SR measures heavily relies on how well the underlying knowledge base can be used to assess semantic relatedness based on term cooccurrence. The Explicit Semantic Analysis (ESA) model [9] showed that Wikipedia is seemingly the most appropriate single resource for this purpose, and later work [34] showed that alternative resources can provide comparable performance (or even a noticeable advantage, e.g. for verb pairs). These results in the literature suggest that the combination of multiple resources can lead to improved performance and more robust behavior across different parts of speech at the same time.

The combination of resources has already been shown to be beneficial for word semantic relatedness [1]. Agirre et al. [1] report the best results so far on the English WS-353 [8] and RG-65 [25] datasets, by combining personalized PageRank on the WordNet graph with the contextual and syntactic dependency profiles of words over a large web-based corpus. These approaches are not straightforward to apply to longer texts, at least without significant extra computation (see above). Thus, here we combine resources using concept vector measures and also employ a thorough extrinsic evaluation of combination using three NLP applications.

3 Concept Vector based Semantic Relatedness

Concept vector based SR methods represent words as a vector of articles in a specific document collection describing world knowledge (each document representing a real world concept). Semantic relatedness is then calculated using a vector similarity function. Formally, for a content word t, the concept vector $\overrightarrow{t}$ is defined as $\overrightarrow{t} = \{w_{c_1 t}, w_{c_2 t}, \ldots, w_{c_n t}\}$, where $w_{c_i t}$ represents the weight of the concept c_i for the word t (e.g. the term frequency of t in c_i) and n is the collection size. The relatedness of terms t_1 and t_2 can thus be calculated using a vector similarity measure, e.g. cosine similarity: $sim_{cosine}(t_1, t_2) = \frac{\sum_i w_{c_i t_1} \cdot w_{c_i t_2}}{\sqrt{\sum_i w_{c_i t_1}^2} \cdot \sqrt{\sum_i w_{c_i t_2}^2}}$.

Following [6, 9], longer text segments can be represented using the centroid of the individual term concept vectors. The relatedness of two text segments can then be determined using the same vector similarity function as on the word level. As a result, it is unnecessary to compute the relatedness between all word pairs in the respective documents to get the document-level relatedness score.

3.1 Concept Vector based SR Parameters

At the core of concept vector based methods is measuring the term cooccurrence statistics over Wikipedia (or a similar resource). The most important technical parameters of such a measure are:

Vector Similarity Function An arbitrary $f(t_1, t_2) \mapsto \Re$ vector similarity function can be used to compare two concept vectors. Thereby, (t_1, t_2) is considered more similar to each other than (t_3, t_4) if $f(t_1, t_2) > f(t_3, t_4)$.

Gabrilovich and Markovitch [9] used the cosine similarity, and recently Hassan and Mihalcea [13] proposed a Lesk-like [18] vector similarity function, and argued that it is more suitable for cross-language relatedness. We assume that the concept vectors are normalized, and simplify the formulas accordingly:

- $sim_{dotprod}(t_1, t_2) = \sum_i w_{c_i t_1} \cdot w_{c_i t_2}$
- $sim_{Lesk}(t_1, t_2) = \sum_i (w_{c_i t_1} + w_{c_i t_2})$, if both $w_{c_i t_1} > 0$ and $w_{c_i t_2} > 0$,

where t_1 and t_2 denote terms, and $w_{c_i t_j}$ denotes the weight of the concept (document) c_i in the knowledge base, for the term t_j.

Component Weights Each concept in the underlying knowledge source has to be assigned a weight in a term's concept vector. The weight is usually defined as a function of the term's frequency in the descriptive text of the concept. Thus, terms that are not used in the descriptive text of a concept are naturally assigned a weight of 0 in the corresponding concept vector. Gabrilovich and Markovitch [9] reported to use *TF.IDF* weights, while Hassan and Mihalcea [13] used a normalized *TF* formula:

- *log TF.IDF*: $w_{c_i t} = log(TF_{c_i t} + 1) \cdot IDF_t$ (the logarithm of the number of times term t appears in document c_i, multiplied by the inverted document frequency of the term in the knowledge base),
- *normalized TF*: $w_{c_i t} = TF_{c_i t} * \log(M/|c_i|)$, where M denotes a constant representing the vocabulary size in the entire knowledge base, and $|c_i|$ represents the vocabulary size of document c_i.

Normalization In concept vector based SR, it is essential to normalize the concept vectors in order to get relatedness values that are comparable to each other. We consider two standard normalization methods: the $L1(\overrightarrow{t}) = \sum_i w_{c_i t}$ and $L2(\overrightarrow{t}) = \sqrt{\sum_i w_{c_i t}^2}$ norms (and divide each vector component by the respective norm value). Even though it is not clearly stated in the literature, we assume that all previous works on concept vector based SR used one of these normalization schemes.

Only the cosine similarity (*dotprod* $L2$) and the ($Lesk$ $L1$) satisfy the criterion that for each term $sim(t, t) = 1.0$. Some combinations, like $Lesk$ $L2$ might even output values larger than 1.0, but this is not important, as long as the scores of the same measure are meaningfully comparable to each other.

Concept Vector Pruning Many studies based on reimplementations of ESA [9] mention that the performance of their system improved greatly when they applied a cutoff threshold and kept just the k highest values in each concept vector, setting very small weights to zero. Gabrilovich and Markovitch [9] employed a pruning threshold defined relative to the highest component weight in the vector (they set all weights to zero when the difference of values in a sliding window of size 100 dropped below 5% of the highest weight), and e.g. Yeh et al. [32] reported to keep just the 625 highest values for English.

	weights	sim.	norm.	pruning	EN	AR	ES	RO	DE
our measure	log-TF · IDF	avgprod	L2	–	**.73**	**.46**	**.51**	**.50**	**.62**
H&M reimpl.	norm-TF	Lesk	L1	–	.49	.28	.26	.29	.50
G&M reimpl.	log-TF · IDF	cosine	L2	–	.61	.25	.21	.24	.52
H&M reimpl.	norm-TF	Lesk	L1	0.01	.70	.43	.43	.43	.60
G&M reimpl.	log-TF · IDF	cosine	L2	0.001	.69	.37	.34	.36	.58
H&M 2009	norm-TF	Lesk	?	?	.71	**.26**	**.50**	**.28**	–
G&M 2007	log-TF · IDF	cosine	L2	sliding w.	**.75**	–	–	–	–
Z et al. 2008	log-TF · IDF	cosine	L2	?	.31-.62	–	–	–	**.65**

Table 1. Spearman rank correlations for different concept vector models on the EN, AR, ES and RO WS353 datasets and the DE Gur350 dataset. '?' indicates a parameter that we could not determine with certainty based on the corresponding papers.

3.2 Our Concept Vector Measure

In our study we used a slightly different concept vector measure with the similarity function: $sim_{avgprod}(t_1, t_2) = \sum_i (w_{c_i t_1} + w_{c_i t_2}) \cdot w_{c_i t_1} \cdot w_{c_i t_2}$, *log TF.IDF* component weights, and *L2* normalization. We chose to use the above implementation, because it does not require the application of an ad-hoc cutoff threshold for term vectors to remove small component weights (i.e. concepts with lower TF values for the given term) in order to show competitive performance. We consider this a positive property as pruning would be inevitably tuned on word relatedness datasets which we also use for evaluation.

3.3 Evaluation on Word Semantic Relatedness

Datasets & Evaluation measures For word semantic relatedness, we use the Spearman rank correlation ρ and the linear Pearson correlation r of SR scores with human judgments as evaluation metrics. Spearman correlation measures how well a monotonic function can describe the relationship between an SR measure and human scores, i.e. how accurately the measure reproduces the relative ordering of word pairs (by humans), while Pearson correlation measures the linear dependence between SR and human scores.

We use publicly available word relatedness datasets for five languages in our experiments. For English, we use the WordSimilarity 353 dataset (EN-WS353) [8]. For German, we use the dataset (DE-Gur350) provided by Gurevych [11]. For Arabic (AR-WS353), Romanian (RO-WS353), and Spanish (ES-WS353), we use the translations of the WS353 dataset provided by Hassan and Mihalcea [13].

Experimental Results In order to compare the proposed vector measure to those used in previous works, we present results on word relatedness in five languages, with Wikipedia as the underlying knowledge resource in Table 1. We compare our measure to those proposed by Gabrilovich and Markovitch [9]

(G&M 2007)[2] and Hassan and Mihalcea [13] (H&M 2009). In order to cope with potential noise due to different preprocessing steps and Wikipedia versions, we provide the results reported in previous works, together with our reimplementation using the same Wikipedia index and preprocessing. In our reimplementation, we employed a pruning threshold relative to the index size (i.e. the number of concepts in Wikipedia for different languages), and kept the k highest values in a concept vector for $k = threshold \cdot index\_size$. For example, for the reimplementation of the H&M 2009 measure, we kept the highest 1% of the concept vector components and set all other weights to 0). We report results without pruning and with pruning (the threshold was fit to provide the best possible result on the EN-WS353 dataset).

As the results in Table 1 demonstrate, our results are in line with the performance scores reported in previous works and our proposed vector measure gives good performance with one less degree of freedom (i.e. no need of tuning a concept vector pruning threshold on the word level, to set small-weight components to zero). This different behavior of the proposed measure can be attributed to the fact that less weight is given to overlapping low-weight vector components (compared to the other vector measures used here). Our results with this configuration are comparable to (with an advantage for languages with smaller Wikipedias) the Spearman correlation values reported using concept vector based SR with Wikipedia for English (0.75) [9]; for German (0.65) [34]; and for Arabic (0.26), Romanian (0.28) and Spanish (0.50) [13]. However, differences in the Wikipedia versions, preprocessing, etc. make direct comparison to previous works difficult, this is why we replicated the corresponding methods. In our subsequent experiments, we use the parameter set described above, i.e. *avgprod* similarity function, *log TF.IDF* component weights, and *L2* norm.

4 Combination of Multiple Resources

For languages where multiple knowledge resources are available, independent concept vector based models can be constructed [34]. We propose the combination of concept vector based SR values based on different resources to construct a measure that performs well across all parts of speech. This would be crucial for a wide range of applications in NLP, and is achieved through the combination of lexical knowledge with the encyclopedic knowledge in Wikipedia. We perform experiments for German and English, using the knowledge resources *Wikipedia*, *Wiktionary*, and *WordNet/GermaNet* for combination.

4.1 Lexical Semantic Knowledge Resources

A lexical semantic knowledge resource provides textual descriptions of concepts from which the concept vectors can be constructed. We use Wikipedia, Wiktionary, and WordNet for this purpose. Before constructing the vector, we pre-

[2] Zesch et al. [34] reimplemented the ESA model [9].

process the textual descriptions using stopword removal and lemmatization (English, German) or stemming (Arabic, Romanian, Spanish).

Wikipedia articles provide detailed textual descriptions for concepts. We used the JWPL Wikipedia API to access the article content. We used the dump from February 6, 2007 (English); September 20, 2009 (Arabic, Spanish); September 19, 2009 (Romanian); February 6, 2007 (German). Following Gabrilovich and Markovitch [9], we discard English Wikipedia articles with less than 100 words and 5 in- or outlinks.

Wiktionary is a multilingual, web-based *dictionary*, *thesaurus*, and *phrase book*, designed as the lexical companion to Wikipedia. In order to get rid of noise from boilerplate text, we used the JWKTL package [34] for fine-grained access to Wiktionary entries. We concatenated the content of all relation types offered by JWKTL for each concept. We used the dump from October 16, 2007 (English) and October 9, 2007 (German).

WordNet [7] and **GermaNet** [16] are lexical databases for English and German. In WordNet, we consider synsets as concept vector components and use the glosses and examples as textual descriptions. As GermaNet contains no glosses, we construct pseudo glosses by concatenating the lemmas of all concepts within a distance of three synsets from the original concept (distance understood as relation path length).

4.2 Combined Concept Vector Measure

A simple and suitable model for combination is to take the individual scores as features, and train regression models to approximate the gold standard scores. For combination, we used the Weka [12] implementations of *Linear Regression (LinReg)* and *Multilayer Perceptron (MLP)* models.

Using regression models, we expect an improved Spearman correlation as the model can learn a nontrivial (and possibly nonlinear) combination of individual values to predict human scores. This setup can also improve the Pearson correlation by seeking an optimal regression model that predicts the human-assigned relatedness values as accurately as possible on the training set.

Datasets & Experimental Setup For English, we used the EN-WS353 dataset [8], the EN-RG65 dataset [25], and the verb relatedness dataset EN-YP130 [31]. For German, we used the translation of the RG65 dataset (DE-Gur65) and the DE-Gur350 dataset [11]. For machine learning experiments, we always used one complete dataset for evaluation. We then performed the training of regression models using the word pairs in the remaining datasets that did not appear in the actual evaluation dataset. For example, for evaluation on the English EN-YP130 dataset, we used the word pairs in the EN-WS353 and EN-RG65 datasets to train the models. This way, our scores are comparable to previous results on these datasets, as they are trained on a disjoint set of word pairs. We did not perform any parameter tuning in order to avoid using parameters that are tuned to the relatively small training sets. Thus, we used the MLP model with 50 training iterations, and all other parameters set to the default values defined by Weka.

Method	EN-WS353		EN-YP130		EN-RG65	
	335		*126*		*65*	
	ρ	r	ρ	r	ρ	r
MLP	.781	**.762**	.701	**.722**	**.860**	**.896**
LinReg	**.790**	.661	.642	.654	.858	.820
Wikipedia	.731	.469	.394	.389	.834	.742
Wiktionary	.661	.390	.628	.462	.803	.569
WordNet	.558	.226	**.715**	.509	.811	.297

Table 2. Spearman rank (ρ) and Pearson (r) correlations (English).

Method	DE-Gur350		DE-Gur65	
	214		*50*	
	ρ	r	ρ	r
MLP	**.774**	**.756**	**.871**	**.891**
LinReg	.769	.679	.870	.809
Wikipedia	.724	.388	.784	.543
Wiktionary	.580	.379	.868	.511
GermaNet	.570	.331	.715	.561

Table 3. Spearman rank (ρ) and Pearson (r) correlations (German).

Word Semantic Relatedness Experimental Results As intrinsic evaluation, we present the results for combining multiple knowledge resources on word relatedness datasets in Tables 2 and 3. We consider this task as intrinsic evaluation as it directly correlates SR scores to human judgments of conceptual similarity. We can assume that the better a measure approximates human scores, the more useful it should be in various NLP applications.

In italics below the dataset name, we show the number of covered word pairs. The first table rows show the results with Multilayer Perceptron (MLP), second rows show Linear Regression (LinReg) for combination. Since all combined measures exploit concept vector based SR on WordNet/GermaNet, Wiktionary, and Wikipedia, we compare them to individual resources (rows 3-5).[3]

As we can see, the use of multiple resources consistently improves over any single resource. Zesch and Gurevych [33] found that increasing the size of the underlying collection (Wikipedia) does not exhibit such remarkable improvements in correlation with human judgments. This confirms our hypothesis that a combi-

[3] The slight differences between values for Wikipedia in Tables 2 and 3 compared to the *'our measure'* row of Table 1 are due to discarding word pairs not covered by Wiktionary or WordNet. This is necessary to ensure a fair comparison of models based on different resources. However, by using only the scores from resources that actually cover the particular word pair, a combined measure with maximal coverage can be constructed. In subsequent extrinsic evaluations, we always employ coverage-maximizing combined measures, assuming 0 values for words not covered by some of the resources.

nation should exploit the advantages of individual resources. The positive effect of the complementarity of the knowledge in different knowledge resources is best demonstrated by the English verb dataset (EN-YP130 column), which was particularly difficult for the otherwise best performing Wikipedia-based measure – the combined measures show a remarkable improvement over the performance of Wikipedia. On the other hand, supervised models perform worse here than WordNet. This is expected, as we used the EN-WS353 and EN-RG65 datasets mostly consisting of noun pairs as the training data, so the models gave more credit to Wikipedia (which performs bad on verbs).

Apart from just one case (EN-YP130 dataset for English) the nonlinear MLP model does not show large improvement over linear regression in terms of Spearman correlation, but it largely improves Pearson correlation. Thus, we suggest the use of linear regression whenever just the ranking of objects is important for an application, as this is the simplest and probably most robust supervised model. The use of a small neural network with sigmoid nodes is a good alternative when one wants to use the measure to retain "similar objects" where "similar" is determined relative to the highest score in a set, as in such settings good Pearson correlation is important.

We consider the application of machine learning to combine the results of concept vector measures built on multiple knowledge resources promising. Our best result on EN-WS353 is competitive to Agirre et al. [1] (0.78), while preserving the favorable aspect of concept vector measures: same formulation for word and text level, and direct applicability to longer texts without the need to compute relatedness scores for all word pairs. As an additional benefit, concept vector based measures – and their combination – return numerical SR scores (not rank positions like the combination proposed by Agirre et al. that learns pairwise preferences and deduces final ranks from the comparison of all pairs). This is required by applications that need to decide whether the confidence in the returned value is sufficient (the top ranked words/documents might still have quite low SR scores).

5 Extrinsic Evaluation & Discussion

To study the beneficial effects of the combined SR measure incorporating heterogeneous knowledge resources, we compare it to single-resource baselines in solving word choice problems, classification of semantic relations, and text similarity computation.

5.1 Word Choice problems

Word Choice problems [15] consist of a target word and four candidate words or phrases. The objective is to pick the one that is most closely related to the target. The relatedness between the target and each of the candidates is computed by a SR measure, and the candidate with the maximum semantic relatedness value is chosen.

Method	English			German		
	acc.	cov.	H	acc.	cov.	H
MLP	.742	**.997**	.851	.740	**.848**	.790
LinReg	.746	**.997**	**.853**	.770	**.848**	**.807**
Wikipedia	.600	**.997**	.749	.718	.821	.766
Wiktionary	.835	.602	.700	**.886**	.313	.463
WN / GN	**.855**	.529	.654	.637	.310	.417

Table 4. Accuracy and coverage in solving word choice problems (English and German).

Datasets & Evaluation Measures In our experiments, we used the datasets introduced by Jarmasz and Szpakowicz [15] of 300 Word Choice (WC) problems for English, and by Zesch et al. [34] of 1008 WC problems for German. We lemmatized the target and all candidates. We employed the standard evaluation metrics for this task, i.e. we measured the accuracy (percent of WC problems solved correctly), coverage (percent of WC problems with all alternatives represented in the knowledge base and at least one with nonzero relatedness) and H (harmonic mean of the accuracy and coverage) of SR measures.

Experimental Results In Table 4, we present results for different measures and their combination in solving word choice problems. The combined measures show very positive characteristics: the coverage is better or equal to the highest coverage of an individual resource, while the accuracy is closer to the most accurate lexical resources than Wikipedia's (which is unpaired in coverage by the other knowledge resources). Overall the combined measure gives an improvement of more than .10 (14% relative increase) in H for English and .04 (5% relative increase) for German. The best results in Table 4 are in the range of the state-of-the-art ($H = .86$ [15] for English and $H = .75$ for German [34]).

5.2 Classification of Semantic Relations between Nominals

The classification of semantic relations between nominals aims at the identification of specific relation types between nouns or base noun phrases appearing in natural language sentences collected from the web. Hendrickx et al. [14] proposed to identify and classify instances of 9 abstract semantic relations between noun phrases, i.e. *Cause-Effect, Instrument-Agency, Product-Producer, Content-Container, Entity-Origin, Entity-Destination, Component-Whole, Member-Collection, Message-Topic*. That is, given two nominals (*e1* and *e2*) in a sentence, systems have to decide whether *relation(e1,e2)* or *relation(e2,e1)* holds for one of the relation types or the nominals' relation is *other* (other relation or unrelated).

Datasets & Evaluation Measures For the classification of semantic relations between nominals, we use the dataset (8000 train and 2717 test sentences) and

standard evaluation measure [14], i.e. the macro averaged F measure for the various relation types. We also provide the classification accuracy scores that serve our goal to compare the effect of SR measures better.

Method	**Train (10 fold)**		**Test**	
	macro F	acc.	macro F	acc.
MLP	**.708**	**.668**	**.689**	**.647**
Wikipedia	.694	.654	.680	.635
Baseline	.657	.621	.605	.561
MLP (no lex.)	**.585**	**.550**	–	–
Wikipedia (no lex)	.558	.524	–	–
Baseline (no lex.)	.373	.385	–	–

Table 5. Macro average F and classification accuracy in relation classification.

Experimental Results For comparison, we employed a baseline system using *standard lexical* (word unigram and lemma uni- and bigrams), *surface* (sentence length, distance of the nouns in tokens), and *contextual* (POS uni-, bi- and trigrams, dependency relations between the nouns) features for classification. To test the added value of semantic relatedness measures, we added SR features to the baseline classifier, describing the relatedness of the nouns to be classified to a set of clue words characteristic for one of the relations (e.g. *goods, cargo, bottle* for *Content-Container*)[4].

In Table 5, we compare the performance of the relation classification system with the baseline features and with extended feature sets using Wikipedia-based SR features and SR features provided by the combined measure. We also compare SR performance without lexical features (i.e. when used in a nonlexicalized classifier). The results show consistent improvements over the Wikipedia-based measure, and huge improvements over the baseline without SR (indicating that SR incorporates useful world knowledge to the classifier model). The best results in Table 5 are in the range of the state-of-the-art performance (0.52-0.82 macro average F measure [14]), with top performance reached using richer representations than the one used here. For details, see Szarvas and Gurevych [28].

5.3 Text Similarity

Two texts are considered similar when their semantic content is closely related to each other. Text similarity computation aims at quantifying the conceptual similarity between two input texts and correlates the calculated similarity scores to the human notion of document similarity through comparison to similarity

[4] The list of clue words, feature set and additional material can be found at http://www.ukp.tu-darmstadt.de/data/sr-combination/.

Method	full		part 1		part 2	
	r	ρ	r	ρ	r	ρ
MLP	.621	**.576**	**.743**	.616	**.757**	.567
LinReg	**.727**	.571	.702	**.630**	.721	**.570**
Wikipedia	.707	.563	.688	.615	.722	.543
Wiktionary	.500	.376	.411	.350	.563	.407
WordNet	.582	.452	.566	.436	.597	.470
G&M reimpl.	.697	.484	.704	.516	.709	.464
G&M 2007	.72	–	–	–	–	–

Table 6. Pearson (r) and Spearman rank (ρ) correlations on the Lee et al. (2005) dataset.

scores assigned by readers. This task has natural applications in information search and content management.

In our text similarity implementation, we use the document-level aggregation based on centroid vectors [9].

Datasets & Evaluation Measures For text similarity experiments, we use the 1225 similarity pairs provided by Lee et al. [17], and similar to previous works we use Pearson correlation for evaluation (and also list Spearman correlation).

Experimental Results In Table 6, we compare the combined measures to single resources and to our reimplementation of ESA [9]. We used the word similarity datasets for training the combined models. This approach has the weakness that supervised models are trained on word similarity scores which have largely different characteristics from text similarity values. For WordNet and Wiktionary, many word pairs receive 0 similarity (i.e. they do not cooccur at all in the small text definitions in these resources), which is seldom the case for document pairs. This difference in the distribution of feature values is easily noticeable in the Pearson correlation of the nonlinear combination. To mimic a more ideal setting, when combined measures are trained on a set of document pairs (with assigned similarity scores), we cut the Lee et al. dataset into two parts and report results on each part (the combined models here are trained on the other half of the dataset). Besides the unexpected behavior mentioned above, we again see a consistent improvement through combination of different knowledge sources, over single resource measures.

These results suggest that multiple knowledge sources serve as a better basis for comparison of the similarity of text pairs. Or to put that in a wider context, the individual SR measures built on different resources would be good separate features for *learning to rank* (where a similar combination of features is performed to develop improved ranking functions) [19], as their improvements add upon each other. The best results in Table 6 are in the range of the state of the art performance of 0.60 [17] to 0.77 [32] Pearson correlation (Spearman is not used by previous studies).

6 Conclusions & Future Work

This paper demonstrated that better and more robust SR measures (that are applicable to single words and texts alike) can be obtained through the combination of concept vector measures exploiting various independent knowledge resources. First, we provided a detailed overview of concept vector based semantic relatedness measures, identified the most important parameters (term weighting, vector similarity function, vector normalization, and concept vector pruning) that can have major effect on the performance of the concept vector model as an SR measure. Thus, future work should state clearly the parameters of the implementation used, or the results will become difficult to reproduce and compare. Moreover, we proposed a formulation that has one less degree of freedom – it does not require the pruning of the word vectors – and performs well for representative datasets in five languages.

Our second main contribution is the combination of concept vector based SR values computed on different underlying resources by means of machine learning. To combine relatedness measures, we used a regression framework that preserves the direct applicability of concept vector based measures to longer texts. We demonstrated that the combination of resources yields stable performance across parts of speech and consistently improves performance in word relatedness over the standard knowledge base (Wikipedia) for concept vector based measures.

Finally, we performed a thorough extrinsic evaluation using three different NLP tasks: solving word choice problems, classification of semantic relations between nominals, and text similarity. We demonstrated that the improved correlation scores of our combined measure on standard word relatedness datasets actually lead to positive effects in all these applications. Thus, these experimental evaluations prove the feasibility of our approach and the hypothesis that through combining heterogeneous knowledge sources for concept vector based semantic relatedness, more robust and accurate measures can be developed that are also applicable to longer texts. The good results in text similarity calculation also suggest that these vector similarity values based on different knowledge sources are promising candidates for separate features in learning to rank (as their positive characteristics can be combined using machine learning).

In future work, we plan to extend our model to incorporate further resources and similarity measures, and to apply these measures together with traditional Information Retrieval similarity functions, in learning to rank [19].

7 Acknowledgements

This work was supported by the German Ministry of Education and Research (BMBF) under grant 'Semantics- and Emotion-Based Conversation Management in Customer Support (SIGMUND)', No. 01ISO8042D, and by the Volkswagen Foundation as part of the Lichtenberg-Professorship Program (I/82806).

References

1. E. Agirre, E. Alfonseca, K. Hall, J. Kravalova, M. Pasca, and A. Soroa. A Study on Similarity and Relatedness Using Distributional and WordNet-based Approaches. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*, pages 19–27, 2009.
2. D. Bollegala, Y. Matsuo, and M. Ishizuka. Measuring semantic similarity between words using web search engines. In *Proceedings of the World Wide Web Conference*, pages 757–766, 2007.
3. A. Budanitsky and G. Hirst. Evaluating WordNet-based Measures of Semantic Distance. *Computational Linguistics*, 32(1):13–47, 2006.
4. H.-H. Chen, M.-S. Lin, and Y.-C. Wei. Novel association measures using web search with double checking. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 1009–1016, 2006.
5. P. Cimiano, A. Schultz, S. Sizov, P. Sorg, and S. Staab. Explicit versus latent concept models for cross-language information retrieval. In *Proceedings of the 21st international jont conference on Artifical intelligence*, pages 1513–1518, 2009.
6. S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.
7. C. Fellbaum. *WordNet: An Electronic Lexical Database*. MIT Press, 1998.
8. L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, and G. Wolfman. Placing Search in Context: The Concept Revisited. *ACM Transactions on Information Systems*, 20(1):116–131, 2002.
9. E. Gabrilovich and S. Markovitch. Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis. In *Proceedings of The 20th International Joint Conference on Artificial Intelligence*, pages 1606–1611, 2007.
10. E. Gabrilovich and S. Markovitch. Wikipedia-based semantic interpretation for natural language processing. *Journal of AI Research*, 34(1):443–498, 2009.
11. I. Gurevych. Using the Structure of a Conceptual Network in Computing Semantic Relatedness. In *Proceedings of the 2nd International Joint Conference on Natural Language Processing*, pages 767–778, 2005.
12. M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The WEKA Data Mining Software: An Update. *SIGKDD Explorations*, 11(1):10–18, 2009.
13. S. Hassan and R. Mihalcea. Cross-lingual Semantic Relatedness Using Encyclopedic Knowledge. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1192–1201, 2009.
14. I. Hendrickx, S. N. Kim, Z. Kozareva, P. Nakov, D. Ó Séaghdha, S. Padó, M. Pennacchiotti, L. Romano, and S. Szpakowicz. Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the 5th SIGLEX Workshop on Semantic Evaluation*, 2010.
15. M. Jarmasz and S. Szpakowicz. Roget's Thesaurus and Semantic Similarity. In *Proceedings of Recent Advances in NLP*, pages 111–120, 2003.
16. C. Kunze. *Lexikalisch-semantische Wortnetze*, chapter Computerlinguistik und Sprachtechnologie, pages 423–431. Spektrum Akad. Verlag, 2004.
17. M. D. Lee, B. Pincombe, and M. Welsh. An empirical evaluation of models of text document similarity. In *Proceedings of the 27th Annual Conference of the Cognitive Science Society*, pages 1254–1259, 2005.

18. M. Lesk. Automatic Sense Disambiguation Using Machine Readable Dictionaries: How to tell a pine cone from an ice cream cone. In *Proceedings of the 5th Annual Int. Conference on Systems Documentation*, pages 24–26, 1986.
19. T.-Y. Liu. Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval*, 3(3):225–331, 2009.
20. D. Milne and I. H. Witten. An effective, low-cost measure of semantic relatedness obtained from wikipedia links. In *Proceedings of WikiAI*, pages 25–30, 2008.
21. M. Mohler and R. Mihalcea. Text-to-text semantic similarity for automatic short answer grading. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 567–575, 2009.
22. C. Müller and I. Gurevych. A Study on the Semantic Relatedness of Query and Document Terms in Information Retrieval. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1338–1347, 2009.
23. S. Padó and M. Lapata. Dependency-based construction of semantic space models. *Comput. Linguist.*, 33(2):161–199, 2007.
24. S. Patwardhan, S. Banerjee, and T. Pedersen. Using Measures of Semantic Relatedness for Word Sense Disambiguation. In *Proceedings of the 4th International Conference on Intelligent Text Processing and Computational Linguistics*, pages 241–257, 2003.
25. H. Rubenstein and J. B. Goodenough. Contextual Correlates of Synonymy. *Communications of the ACM*, 8(10):627–633, 1965.
26. M. Stevenson and M. Greenwood. A Semantic Approach to IE Pattern Induction. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pages 379–386, 2005.
27. M. Strube and S. P. Ponzetto. WikiRelate! computing semantic relatedness using wikipedia. In *Proceedings of the 21st AAAI National Conference on Artificial Intelligence*, pages 1419–1424. AAAI Press, 2006.
28. Gy. Szarvas and I. Gurevych. Tud: Semantic relatedness for relation classification. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 210–213, 2010.
29. G. Tsatsaronis and V. Panagiotopoulou. A Generalized Vector Space Model for Text Retrieval Based on Semantic Relatedness. In *Proc. of the Student Research Workshop at EACL 2009*, pages 70–78, 2009.
30. S. Wubben and A. van den Bosch. A semantic relatedness metric based on free link structure. In *Proceedings of the 8th International Conference on Computational Semantics*, pages 355–358, 2009.
31. D. Yang and D. M. W. Powers. Verb Similarity on the Taxonomy of WordNet. In *Proceedings of the 3rd International WordNet Conference*, pages 121–128, 2006.
32. E. Yeh, D. Ramage, C. D. Manning, E. Agirre, and A. Soroa. WikiWalk: Random walks on Wikipedia for Semantic Relatedness. In *Proceedings of the Workshop on Graph-based Methods for Natural Language Processing*, pages 41–49, 2009.
33. T. Zesch and I. Gurevych. The more the better? assessing the influence of wikipedias growth on semantic relatedness measures. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation*, 2010.
34. T. Zesch, C. Müller, and I. Gurevych. Using Wiktionary for Computing Semantic Relatedness. In *Proceedings of the 23rd AAAI Conference on Artificial Intelligence*, pages 861–867, 2008.