Philipp Scholl, Doreen Böhnstedt, Renato Domínguez García, Christoph Rensing, Ralf Steinmetz
Extended Explicit Semantic Analysis for Calculating Semantic Relatedness of Web Resources
In: M. Wolpers, P. Kirschner, M. Scheffel, S. Lindstädt, V. Dimitrova (eds.): Sustaining TEL: From Innovation to Learning and Practice
Proceedings of EC-TEL 2010, Lecture Notes in Computer Science 6383, p. 324-339, Springer Verlag, September 2010. ISBN 978-3-642-16019-6.

Extended Explicit Semantic Analysis for Calculating Semantic Relatedness of Web Resources

Philipp Scholl, Doreen Böhnstedt, Renato Domínguez García,
Christoph Rensing, and Ralf Steinmetz

Multimedia Communications Lab (KOM)
Technische Universität Darmstadt
Rundeturmstr. 10, 64283 Darmstadt, Germany
{scholl,boehnstedt,renato,rensing,ralf.steinmetz}@kom.tu-darmstadt.de
http://www.kom.tu-darmstadt.de

Abstract. Finding semantically similar documents is a common task in Recommender Systems. Explicit Semantic Analysis (ESA) is an approach to calculate semantic relatedness between terms or documents based on similarities to documents of a reference corpus. Here, usually Wikipedia is applied as reference corpus. We propose enhancements to ESA (called Extended Explicit Semantic Analysis) that make use of further semantic properties of Wikipedia like article link structure and categorization, thus utilizing the additional semantic information that is included in Wikipedia. We show how we apply this approach to recommendation of web resource fragments in a resource–based learning scenario for self–directed, on–task learning with web resources.

Key words: Explicit Semantic Analysis, Semantic Relatedness, Wikipedia, Reference Corpus, Recommendation

1 Introduction and Motivation

A common task in Information Retrieval is finding documents that are similar to a given document. *Similarity* in this context has been usually determined as a measure of term overlap that occurs in these documents [1]. However, in recent work, a more high–level measure called *semantic relatedness* that abstracts from the terminology used and aims towards a more semantic dimension, where the similarity between *concepts* of the underlying documents is taken into account.

This is especially useful as humans tend to focus the similarity of documents in concepts rather than in terms. Especially in domains where users need to find similar documents but do not exactly know the terminology, abstracting from terminology towards a more semantic measure can be applied.

One of those domains is the domain of Technology Enhanced Learning (TEL), where different audiences with different levels of knowledge exists. For example, novices tend to be not aware of terminology of the domain they are learning,

whereas experts are able to communicate in a brief manner using the professional terminology. Further, in different stages of achieved expertise, different aspects of learning materials are important, giving either a broad overview or rather a very narrow scope of the learning domain.

Thus, for applications in TEL that support retrieval and recommendation of documents, being able to find semantically related documents is an essential task.

1.1 Crokodil

The scenario we address with this work is a research prototype for supporting self–directed, resource–based learning with web resources, *Crokodil* (a project based on ELWMS.KOM [2]). As the importance of the World Wide Web as a major source for knowledge acquisition has been growing steadily in the last decade, both specifically designed learning materials (e.g. information contained in Web Based Trainings or tutorials) and web resources that are not specifically intended to be used for learning (e.g. user generated content in blogs, wikis or forums) are available at a large scale.

Crokodil supports learners in finding, collecting and organizing these learning materials in a so–called *Knowledge Network* (KN). These KN are based on Semantic Networks [3], that represent knowledge in a graphical notation consisting of nodes and relations. In *Crokodil*, the learning materials are stored as nodes in the KN. A peculiarity is that often only a part of a web resource is relevant for the information needs of a learner. *Crokodil* allows saving only the needed fragments, furthermore called *snippets*.

A major challenge is supporting learners using *Crokodil* in finding learning materials that are relevant for their current information needs. Therefore, we propose a *recommender system* that helps learners finding related content from other learners by recommending snippets that are semantically related to those they recently added to their KN.

1.2 Snippets

In a user study [4], we evaluated — among other research questions — how learners select relevant content. The lab study served to examine how learners can be supported in organizing their learning processes with web resources by setting goals. During the study, participants were asked to collect learning materials from web resources, learn with the assembled information and take a performance test afterwards. The participants were instructed to collect the information from the web resources that they deemed to be relevant for their learning tasks, allowing them to select content in the *desired granularity*. Thus, we collected 1357 different snippets from 104 participants.

For comparing the properties of snippets with "normal" bookmarked web pages (as these serve a similar goal), we randomly crawled Delicious[1], a social

[1] http://delicious.com

bookmarking service that allows storing relevant URLs online, for a comparison corpus. We downloaded 1004 HTML pages thereof and, after stripping HTML–specific content like markup, compared them to the snippets gained from our study.

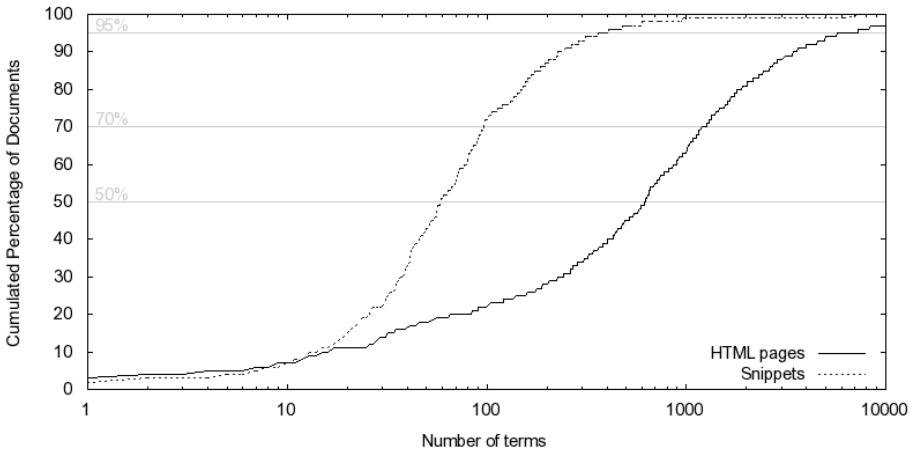

Fig. 1. Cumulated term counts of snippets in comparison with term counts of full web pages

The results (see fig. 1) show, that snippets differ from whole web resources in some accords:

- They mostly deal with a specific, well–defined domain, covering only one subject. Web pages, however, usually cover a lot more information.
- On average, snippets consist of 120 terms, whereas web pages consist of about 1600 terms.
- 70% of snippets are smaller than 100 terms, 70% of web pages are smaller than 1000 terms.

Based on observations in this analysis of snippets, we state the requirements an approach should have in order to generate content–based recommendations:

- In short snippets, there are only few significant terms. A larger context is not available, thus the algorithm will have to abstract from the term level.
- The algorithm should be stable and provide good results, no matter how long the different snippets are.
- The snippets may be about any topic. Thus, the algorithm should be able to infer over any generic knowledge domain.
- Learners should be able to inspect *why* two snippets are regarded as being semantically related. This allows the learners to analyze if the recommended item is really relevant in their current learning situation.

The remainder of this paper is organized as follows: in section 2, we give an overview of related work, map it with our requirements, and present the foundation of our work, Explicit Semantic Analysis (ESA). In section 3, we introduce our extensions to the basic ESA approach. Selected evaluations of our approach are presented in section 4. Eventually, we conclude in section 5 and give an outline for next steps and open issues.

2 Related Work

Most approaches to compare documents apply the vector space model (VSM) [1] in combination with the cosine similarity for calculating document similarity. Thus, approaches based on VSM have in common to quantify the syntactic overlap. Documents are represented by high–dimensional feature vectors derived from the terms used in the document. The similarity between two documents is modeled by the angle between the representing vectors. However, as the vectors are entirely based on syntactical features, i.e. the term occurrences in the document, VSM is not applicable in cases of documents that are semantically related but have little term overlap.

Further, in some scenarios similarity should not be expressed over terms but over the meaning of a document. For example, different documents written for or by differing audiences (e.g. beginners vs. experts) may be written using a different terminology, e.g. using synonyms or hypernyms. Although these documents describe the same semantic concepts, the term–based similarity will be rather low. This is called the *vocabulary gap* [5].

Therefore, there is a need to abstract from the terms used in a document towards a more semantic representation, meaning that similarity is not to be expressed via common terminology, but rather by usage of terminology in a common semantic context. As *similarity* is a term that is not really applicable to the semantic dimension, the term usually preferred is *semantic relatedness* [6]. In this work, we use the term *similarity* when a term–based measure is applied and *relatedness*, when a semantic measure is applied.

As semantic relatedness is a measure that is — at least — difficult to calculate with only the documents to compare, related approaches usually utilize additional information by employing *reference corpora* in order to provide additional general knowledge. In related work, many different corpora have been used. Most provide structured access to semantic properties of terms (e.g. WordNet, Roget's Thesaurus), whereas other corpora, like Wikipedia, represent the underlying semantics inherently in the documents they contain.

One of the most popular reference corpora is WordNet [7], a broad coverage lexical network of English words. WordNet provides networks of synsets that contain terms like nouns, verbs, adjectives and adverbs, each representing a lexical concept. The synsets are interlinked with a variety of relations (e.g. denoting homonymy, synonymy, etc.). Jiang and Conrath [8] combine an approach using WordNet with corpus statistics. They merge a content–based, node–centric information content approach with a node–distance, edge–centric approach and

apply those to the WordNet noun synsets. According to [6], this approach performs better than others they compared. Another popular data source that has been used for calculating semantic relatedness is Roget's Thesaurus. Jarmasz and Szpakowicz [9] use it as a base to calculate *semantic distance* between terms based on the path length in the thesaurus graph. They convert the distance to semantic similarity by subtracting the path length from the maximum possible path length.

However, both corpora are well–structured and have to be manually serviced by experts. Roget's Thesaurus, for example, dates from 1805 (with an edition from 1911 in the public domain). Although general terminology is contained, the Thesaurus cannot keep up with the rapid evolution of knowledge nowadays.

Another approach that has gained momentum in the last years is Latent Semantic Analysis (LSA) [10]. LSA is an approach that uses a custom corpus of documents to abstract from the terminology used and derives inherent semantic concepts. So, with LSA, different terms that are used as synonyms or are cooccurring often are mapped into a single concept. Further, by mapping terms, the overall corpus dimensions may be reduced significantly, thus transforming the search space. This projection and reduction is achieved by applying a singular value decomposition on a corpus matrix and then truncating the least significant values. LSA, although being a stable approach that performs well, has some limitations regarding our requirements stated in section 1. First, the approach needs to be given the dimensions to reduce. This heavily depends on the topics of the documents that are present in the scenario. Second, the resulting concepts are sets of terms and are not easily readable by humans. Thus, we refrain from applying it to our scenario.

In recent research, the collaboratively edited, open encyclopedia Wikipedia has been increasingly used for information retrieval related tasks (e.g. [11], [12] and [13]). This is due to the sheer amount of articles available (over 1 Mio articles in the German version as of 2010), with each article representing a distinct *concept*. Additionally, Wikipedia provides further semantic information about the concepts described in articles, most notably links to related articles (*article links*) and a (mostly hierarchical) category structure (*category links*). Another criterion that makes use of Wikipedia for information retrieval suitable is that it is constantly updated to the current state of knowledge, e.g. new articles are added and old ones are adjusted accordingly.

WikiRelate! [11] is an approach that computes semantic relatedness between terms. Given two terms to analyze, WikiRelate! searches the Wikipedia article names (called *lemmata*) for the terms and calculates the distances between found articles based on the articles' contents and the category structure of Wikipedia. As it only supports computation of semantic relatedness between terms, this approach is not applicable to documents.

Kaiser et al. [14] introduce *conceptual contexts* of documents as linkage graphs that represents the document and its relations. Basically, they map the documents to compare to Wikipedia articles and apply a weighting function that determines the article's relatedness to neighbouring articles based on in– and

outgoing article links. After removing all concepts that are only loosely related, they calculate the relatedness measure of the documents by computing the similarity of the graphs.

2.1 Explicit Semantic Analysis

A promising approach to calculating semantic relatedness named *Explicit Semantic Analysis (ESA)* [15] has been proposed by Gabrilovich and Markovitch. Here, documents are not represented by occurring terms but by their similarity to concepts derived from Wikipedia articles. ESA is based on the assumption that in Wikipedia an article corresponds to a semantically distinct concept. Thus, by comparing documents to all articles in a Wikipedia corpus that has been pre–processed by tokenization, stemming, stop word removal and a term weight metric, a vector is obtained that contains a similarity value to each of the articles. This vector, called *semantic interpretation vectors*, abstracts from the actual term occurrences and thus represents a semantic dimension of that document. A major advantage of ESA is that semantic relatedness can be calculated for terms and documents alike, providing good and stable results for both modes [15].

Formally, the document collection is represented as a $n \times m$ Matrix M (called *semantic interpreter*), where n is the number of articles and m the number of occurring terms in the corpus. M contains (normalized) *tf–idf* document vectors of the articles. *tf–idf* is a commonly used measure of relevance of a term in relation to a corpus D, where the *term frequency* tf of term t_i for each document $d_j \in D$ and the *inverse document frequency idf* of all occurrences of term t_i are taken into account. For calculating the similarity between the document and the corpus, the *cosine similarity measure* (1) is employed. Analogously, two documents represented as semantic interpretation vectors can be easily compared by using cosine similarity again.

$$sim(d_i, d_j) = \cos(\phi) = \frac{d_i \cdot d_j}{|d_i| * |d_j|} \tag{1}$$

Gabrilovich and Markovitch show that ESA outperforms other approaches like WikiRelate!, WordNet, Roget's Thesaurus and LSA [15]. Kaiser et al. [14] see ESA as a competitor to their approach using conceptual contexts, but they do not compare their approach to ESA.

Although ESA is commonly used with Wikipedia as reference corpus, it is not necessarily restricted to it. In theory, all textual corpora that follow the structure of providing unique documents (i.e. covering different topics) could be applied. Gabrilovich and Markovitch [15] apply ESA to a corpus derived from the Open Directory Project themselves, mapping concepts to the categories of the directory. Notably, Anderka and Stein [16] dismiss the hypothesis that the reference corpus needs to be semantically well–structured, i.e. semantic concepts are only described by one document. They show that ESA with the Reuters newswire corpus and even random corpora may achieve comparable results to

ESA with Wikipedia. Still, as Wikipedia provides distinct semantic concepts as labels (i.e. the lemmata of the articles), it is better for humans to interpret and understand the relatedness between documents.

In general, ESA fulfills the requirements as a foundation for our recommendation algorithm stated in section 1. ESA can cope with documents of arbitrary size, has the backing of a broad knowledge base (i.e. Wikipedia) and performs well compared to other approaches. Still, it leaves space for improvements, especially as far as utilization of additional semantic information from the Wikipedia corpus goes. Thus, in the next section, we will present our adjustments to ESA.

3 Our Approach

As described above, Explicit Semantic Analysis (ESA) only makes use of the article information that Wikipedia contains, i.e. only analyzes the term $\rightarrow$ article allocation. However, Wikipedia provides a wealth of semantic information, namely the links between articles and the categorization structure of articles. ESA neglects this available information completely.

Thus, we introduce *eXtended Explicit Semantic Analysis* (XESA), an approach that semantically enriches the interpretation vectors obtained from ESA, which has been described in section 2. In detail, we present three different approaches to extending ESA, one utilizing the article link graph of Wikipedia, one using the category structure and one approach that combines those two. We expect these approaches to perform better than ESA, as they enrich ESA by additional semantic information.

However, before presenting the XESA extensions, we revisit ESA formalizing our approach to implementing it:

- First, we preprocess a Wikipedia dump[2] (in our work we use the dump of the German Wikipedia from June 2009) with stemming, stop–word removal, *tf–idf* calculation and normalization.
- Then, we aggregate all article vectors into the semantic interpreter matrix M with the shape $n \times m$, where n is the number of articles and m the number of terms.
- For each document d that is to be compared, the same preprocessing steps have to be executed, so that we receive the document vector v_d with the form $1 \times m$, where m is the number of terms.
- As all document vectors are normalized, we get the interpretation vector i_{esa} that represents the cosine similarity of v_d with all article vectors of M simply by applying the inner product (2) with transposed M.
- Finally, the result is the interpretation vector i_{esa} with the dimensions $1 \times n$.

$$i_{esa} = v_d \cdot M^T \tag{2}$$

[2] available from `http://dumps.wikimedia.org/`

This interpretation vector i_{esa} is the foundation for all further approaches. The basic idea is to enrich this interpretation vector with additional information derived from semantic information that can be extracted from the Wikipedia corpus.

3.1 Utilization of the Article Graph

On average, Wikipedia articles link to 30 other articles. These links can be interpreted as semantic relationships to other concepts. For example, the German article for *General Relativity* links to other articles *Space*, *Time* and *Gravitation*. Thus, there is an obvious generic relatedness to the concepts expressed by these article links.

The overall article linkage graph of Wikipedia can be represented as the adjacency matrix $A_{Articlegraph}$ of dimensions $n \times n$, where n is the number of articles. If an article a_i links to a_j, the respective cell in the matrix is set to 1, otherwise it is set to zero, resulting in a sparse matrix.

$$A_{Articlegraph_{i,j}} = \begin{cases} 1 & \text{if } a_i \text{ contains a link to } a_j \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

Initially, we intended to include weights that decrease with the linkage distance of articles on indirect links, e.g. if a_i links to a_j and a_j links to a_k, that a value greater than 0 (but less than 1) is inserted into $A_{Articlegraph_{i,k}}$. Yet, preliminary tests showed that the semantic relatedness between articles linked by second degree is already very low, thus it would only raise computation overhead without contributing to the result. Therefore, we refrained from adding this weighted distance measure.

As articles never contain a reference to themselves, the adjacency matrix has to be added to the identity matrix $I_{|articles|}$ so that the diagonals are not 0. Otherwise, there is the possibility that already computed information is lost. Further, a weight factor w is introduced that determines how strong the influence of the article graph is on the original i_{esa}. On multiplication of the semantic interpreter from ESA with the resulting matrix (4), the relevant semantic information already present in i_{esa} is reinforced.

$$i_{xesa:ag1} = i_{esa} * (w * A_{Articlegraph} + I_{|articles|}), w \in [0..1] \tag{4}$$

Performance–wise, the article graph extension poses the challenge that the complete interpretation vector has to be multiplied with a large matrix again. Additionally, we observed i_{esa} to usually contain only few similarity values that are significant and lots of values that are really small. Thus, for boosting efficiency of calculation, we introduce the function `selectBestN` that truncates i_{esa} after the first best N similarity values. This has the effect that the second matrix multiplication is more efficient to be calculated because i_{esa} is sparsely filled with values > 0. Thus, we define a second approach that reduces the overall calculation complexity by only taking the N highest similarity values into account (5).

$$i_{xesa:ag2} = i_{esa} + \texttt{selectBestN}(i_{esa}) * (w * A_{Articlegraph}), w \in [0..1] \quad (5)$$

A challenge, though, is finding an appropriate N that speeds up calculation without deteriorating the quality of the result too much. This will be dealt with empirically in section 4.

3.2 Utilization of Category Information

The category structure of the German Wikipedia contains approximately 80.000 categories with 920.000 articles categorized (approximately 87% of all articles). Besides adminstrative categories and categories that group different articles by properties of the underlying concepts (e.g. *list of German authors by birth year*), there are categories that provide semantic information. These categories represent groupings by semantic properties and express mostly *is–a* relations.

Similar to [17], we append information that encodes category affiliation to the interpretation vector i_{esa}. Therefore, we create the matrix C with the dimensions $n \times m$, where n is the number of articles and m the number of categories (6). On multiplying i_{esa} with C, the result is the vector c_{cat} that encodes information about articles and categories (7).

$$C_{i,j} = \begin{cases} 1 & \text{if article } a_i \text{ links to category } c_j \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

$$c_{cat} = i_{esa} * C \quad (7)$$

$$i_{xesa:cat} = (i_{esa}, c_{cat}) \quad (8)$$

Finally, c_{cat} is appended to i_{esa}. In (8), this is expressed by the appending operator ",". Thus, this operation changes the dimensions of the vector i_{esa} by appending the category vector dimensions. Analogue to the approach using the article graph, this calculation is inefficient if all non–zero values are kept; thus, we apply the above–mentioned `selectBestN` to i_{esa} again, resulting with (9).

$$i_{xesa:cat} = (i_{esa}, \texttt{selectBestN}(i_{esa}) * C) \quad (9)$$

3.3 Combination of Article Graph and Category Extensions

Finally, the article link and category extensions to ESA can be applied in combination. This is rather straight–forward, instead of i_{esa} the result of the article graph extension $i_{xesa:ag1}$ is used (10).

$$i_{xesa:combination} = (i_{xesa:ag1}, \texttt{selectBestN}(i_{xesa:ag1}) * C) \quad (10)$$

We include this approach just for sake of completeness, as the efficiency of this extension is not adequate as will be seen in section 4.

4 Evaluation

In this section, we present an evaluation that compares ESA to our different XESA variants.

4.1 The Snippet Corpus

In order to evaluate XESA, we needed an evaluation corpus that fulfills several requirements:

- In our experiments, we applied the German Wikipedia. Thus, the evaluation corpus should consist of German documents.
- Documents in the evaluation corpus should conform to our snippet definition, i.e. documents should contain between 20 and 200 terms.
- Documents in the evaluation corpus should honour our scenario of resource–based learning with web resources. That is, they should contain a narrow scope of topics and be basically appropriate to answer special information needs.
- Documents should contain different topics and have different scopes, i.e. should not only represent narrow factual knowledge but also contain opinions and overview information.

Because we did not find an appropriate available set of documents that met our requirements, we built a small corpus in a user study [18] with eight participants. The participants were asked to research answers to a catalogue of ten questions using only fragments of web resources. For each question they were to find five snippets that (partially) contained the answer to this question. Further, they were instructed to restrict the snippets' length to 20 to 200 terms. This was not a fixed requirement though, if needed, the participants were allowed to collect larger web resource fragments.

In order to conform to the fourth requirement named above, the questions were formulated so that five different types of questions were asked with two questions per type. We identified the following question types as relevant for our scenario:

- *Opinions*, e.g. "Is the term *Dark Ages* justified?"
- *Facts*, e.g. "What is the FTAA?"
- *Related snippets* to a common topic, e.g. "Find examples for internet slang!"
- *Homonyms*, e.g. "What are Puma, Jaguar, Panther, Tiger and Leopard?"
- *Broad topics*, e.g. "Find information about the evolution of man!"

After having collected the answers, duplicate answers and answers from the same sources were discarded. Finally, the evaluation corpus D consisted of 282 snippets (a short summary is available in table 1) that were labeled with their question types and manually split into groups of different semantic concepts. Because, as expected, homonyms and broad topics showed to be consisting of snippets with different meaning, we got 14 different semantic groups.

Table 1. Short summary of evaluation corpus

Size of corpus	282 documents
Average length of snippets	95 terms
Minimal length	5 terms
Maximum length	756 terms
Standard deviation	71.3 terms

4.2 Evaluation Methodology

For evaluating XESA, we applied a methodology similar to [19] that is used to evaluate search engine rankings. Basically, a semantic relatedness value is calculated for each snippet document $q \in D$ with all $d \in D \setminus q$. The result is a list that is ranked by decreasing relatedness. We define that q and a compared document d_k at rank k are semantically related (i.e. $r_k = 1$) if they are in the same semantic group D_q (11), i.e. they handle the same semantic concept.

$$r_k = \begin{cases} 1 & \text{if } q \text{ and } d_k \in D_q \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

Further, *precision at rank* (12) and *recall at rank* (13) are used to calculate the *average precision* (14) over one relatedness comparison for different recall values. Eventually, all pair–wise comparisons are averaged and the average precision is plotted against recall. One measure that expresses the quality of these results is the *break–even point* [20], the point where precision equals recall (and, as presented in our plots, the interpolated precision–recall curve crosses $f(r) = r$, i.e. the angle bisector of the first quadrant).

$$precision(k) = \frac{1}{k} \sum_{1 \le i \le k} r_i \tag{12}$$

$$recall(k) = \frac{1}{|D_q|} \sum_{1 \le i \le k} r_i \tag{13}$$

$$\text{average precision} = \frac{1}{|D_q|} \sum_{1 \le k \le |D|} r_k * precision(k) \tag{14}$$

For ESA (fig. 2), the break–even point is at 0.575, the mean average precision is 0.595 with standard deviation 0.252.

4.3 Empirical evaluation of selectBestN and Article Graph Weight

As described in section 3, we introduced the function `selectBestN` that discards all i_{esa} values but the n best values for better calculation performance. After some preliminary experiments, we decided to compare the XESA variant using the article graph ($i_{xesa:ag2}$) using three different values, i.e. $n \in (10, 25, 100)$.

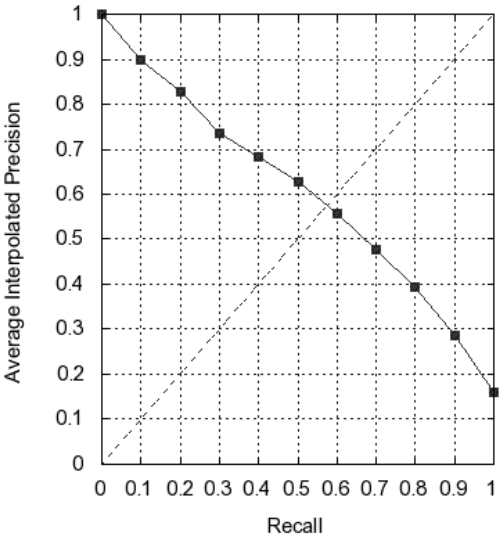

Fig. 2. The precision–recall diagram for basic ESA with the break–even point where $f(r) = r$

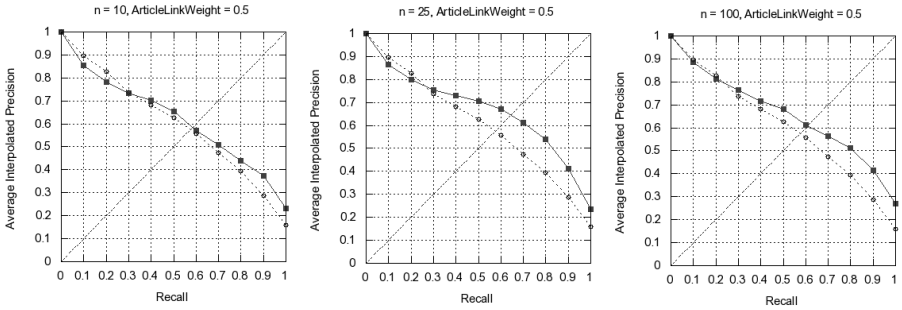

Fig. 3. Calculating the relatedness using the article graph extension with $n \in (10, 25, 100)$

The results in fig. 3 show that the article graph extension performs best with $n = 25$. This is consistent with the results we got using the other extensions as well, so, in the following, we only present results that were computed with $n = 25$.

Further, the article graph weight w used with all XESA article graph extensions was tested with $w \in (0.25, 0.5, 0.75)$. In our experiments, there was no difference between using the weights 0.5 and 0.75. Therefore, we will use $w = 0.5$ in all presented results.

4.4 Comparison of ESA with XESA

In this section, we compare results of the different XESA variants presented in section 3.

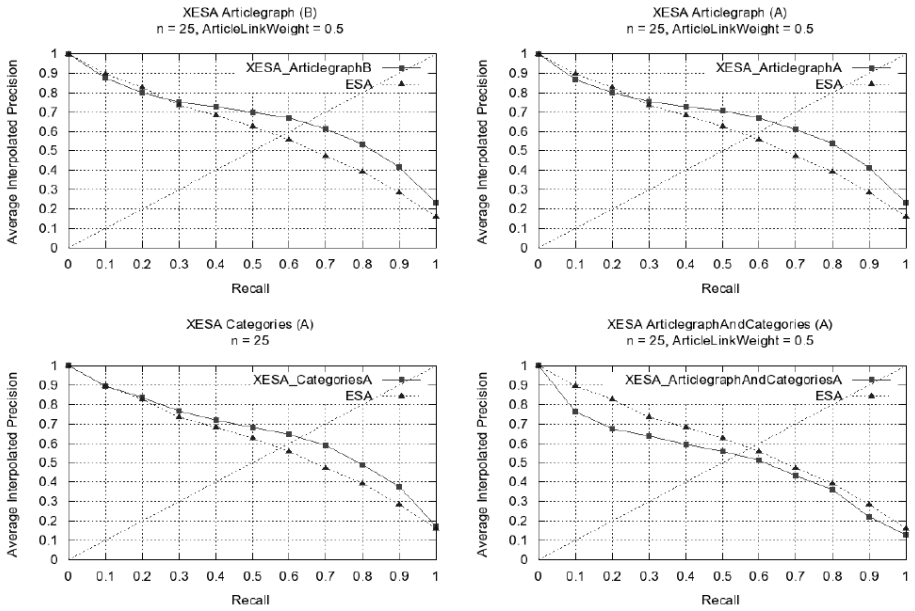

Fig. 4. Precision–Recall Plots of all XESA variants

The precision–recall diagrams of all XESA variants presented in section 3 using the `selectBestN`–parameter n with 25 and the link article graph weight w as 0.5 are displayed in fig. 4. This plot shows that both article link graph extensions perform best, significantly surpassing ESA results by 7%, whereas the category extension still outperforms ESA by 5.4% but cannot measure up to the article graph variants. Both variants combined, however, are not able to even achieve the performance of the basic ESA approach. Detailed results are additionally displayed in table 2.

Table 2. Summary of XESA's results (best are marked bold)

Approach	Break–even Point	Mean Average Precision	Standard Deviation
ESA	0.575	0.595	**0.252**
$\text{XESA}_{xesa:ag1}$	0.644	0.654	0.286
$\text{XESA}_{xesa:ag2}$	**0.645**	**0.657**	0.284
XESA_{cat}	0.629	0.646	0.274
$\text{XESA}_{combined}$	0.539	0.515	0.301

These results show that the semantical information that can be derived from the Wikipedia article graph and the categories is beneficial for computing the semantic relatedness between documents. We think that the article graph variants of XESA perform best because they represent a specific relatedness between concepts. By linking articles, the human editors wanted to express closeness of the underlying concepts. While being linked, some context of this relation can also be found in the linking article as well. For example, the article *General Relativity* links to the article *Space* and shares terminology with that article. Thus, by adding information about the relation, semantic information already known is strengthened by this connection. Categories, however, provide an organizational, top–down view on the concepts. While they provide semantic information about the grouping of articles, they are already abstracted from the specific concept itself. Therefore, the results of XESA's category variant improve ESA but still cannot measure up to the article graph variants.

Further, we presume that the results of the XESA combination variant are worse than ESA's results, because a multiplicatory effect comes into effect. By multiplying the interpretation vectors of different semantic dimensions in that approach, there seems to occur a semantic diversification, i.e. the interpretation vector $i_{xesa:combination}$ is enriched by semantic information based on heterogeneous sources (article graph and categories). Thus, noise is added and the specifity of the semantic dimensions is decreased significantly.

As expected, the 14 semantic groups of the corpus proved to perform differently based on their abstraction. For example, snippets containing fact knowledge in a narrow topic are more easily related than broad topics, because certain terms are common in that group. XESA showed to outperform ESA in recognizing the semantic relatedness between documents in the groups that use different terminology.

Additionally to the evaluation presented here, we compared XESA to ESA in regards of semantic relatedness of single terms. We performed some tests with a corpus created from ratings of the perceived relatedness of 65 term pairs [21]. XESA's results were not significantly different from the same evaluation using ESA. We think that this scenario does not benefit from our approach of semantically enriching the resulting interpretation vector, because the context that is given by additional terms in documents is necessary to exploit the semantic information contained in Wikipedia. Presumably, additional information seemed to add noise to the interpretation vectors that degraded our results.

5 Conclusions and Further Work

In this paper, we presented a scenario of resource–based learning using web resources. We briefly described our research prototype *Crokodil* that aims to support this self–directed way of learning and proposed a recommendation mechanism based on several requirements. We analyzed related work on the basis of these requirements, identifying the approach Explicit Semantic Analysis (ESA) as a foundation for enhancement. We described three approaches of semantically enriching the interpretation vectors obtained by ESA based on Wikipedia article links and categories. Eventually, we evaluated these extensions, subsumed under the name eXtended Explicit Semantic Analysis (XESA), and showed that the extension based on the article link graph, outperforms ESA by 7% on a corpus of snippets. We infer that ESA, albeit a stable and well–performing approach, can be enhanced by using semantic information contained in Wikipedia.

In future work, we will focus on the recommendation engine that provides semantically related content. An open question is, whether and how learners benefit from the offering of unknown, but related, snippets. We think that an interesting research question will be, whether learners profit more from strongly related snippets or weakly related snippets. This requires further evaluations in an open self–directed learning setting. Further, we want to pursuit the question whether Wikipedia lemmata — the titles of the articles — may serve as human–readable topical hints respectively tags for learners. Further we believe that taking into account the relevance of links between articles will improve the article graph extension. For example, *General Relativity* links to *Baltimore*, which is less relevant than the link to *Spacetime*.

A valuable extension to *Crokodil* would be to recommend snippets in other languages that represent the same concepts. As Wikipedia provides inter–language links between articles about the same concepts, this seems to be feasible.

Acknowledgments. This work was supported by funds from the German Federal Ministry of Education and Research under the mark 01 PF 08015 A and from the European Social Fund of the European Union (ESF). The responsibility for the contents of this publication lies with the authors.

References

1. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval. Addison-Wesley Reading, MA (1999)
2. Böhnstedt, D., Scholl, P., Benz, B., Rensing, C., Steinmetz, R., Schmitz, B.: Einsatz persönlicher Wissensnetze im Ressourcen–basierten Lernen. In Seehusen, S., Lucke, U., Fischer, S., eds.: DeLFI 2008: 6. e–Learning Fachtagung Informatik. Number P–132 in Lecture Notes in Informatics, Köllen, Bonn, Gesellschaft für Informatik, Lecture Notes in Informatics (LNI) (Sep 2008) 113–124
3. Sowa, J.F.: Semantic Networks. In Shapiro, S.C., ed.: Encyclopedia of Artificial Intelligence. Volume 2., John Wiley, New York (1992) 1493–1511

4. Scholl, P., Benz, B.F., Böhnstedt, D., Rensing, C., Schmitz, B., Steinmetz, R.: Implementation and Evaluation of a Tool for setting Goals in self-regulated Learning with Web Resources. In Ulrike Cress, Vania Dimitrova, M.S., ed.: Learning in the Synergy of Multiple Disciplines, EC-TEL 2009. Volume LNCS Vol 5794. (2009)
5. Zesch, T., Müller, C., Gurevych, I.: Extracting lexical semantic knowledge from Wikipedia and Wiktionary. In: Proceedings of the Conference on Language Resources and Evaluation (LREC). (2008)
6. Budanitsky, A., Hirst, G.: Evaluating Wordnet-based measures of lexical semantic relatedness. Computational Linguistics **32**(1) (2006) 13–47
7. Fellbaum, C.: Wordnet: An Electronic Lexical Database. MIT Press (1998)
8. Jiang, J.J., Conrath, D.W.: Semantic similarity based on corpus statistics and lexical taxonomy. In: Proceedings of International Conference Research on Computational Linguistics (ROCLING X). (1997)
9. Jarmasz, M., Szpakowicz, S.: Rogets thesaurus and semantic similarity. Recent Advances in Natural Language Processing III: Selected Papers from RANLP (2004) 111
10. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R.: Indexing by Latent Semantic Analysis. Journal of the American society for information science **41**(6) (1990) 391–407
11. Strube, M., Ponzetto, S.P.: Wikirelate! Computing semantic relatedness using Wikipedia. In: Proceedings of the National Conference on Artificial Intelligence. Volume 21., Menlo Park, CA; Cambridge, MA; London;, AAAI Press; MIT Press (2006) 1419ff
12. Milne, D., Witten, I.H.: Learning to link with Wikipedia. In: CIKM '08: Proceeding of the 17th ACM conference on Information and knowledge management, New York, NY, USA, ACM (2008) 509–518
13. Zesch, T., Gurevych, I.: Analysis of the Wikipedia category graph for NLP applications. In: Proceedings of the TextGraphs-2 Workshop (NAACL-HLT 2007). (2007) 1–8
14. Kaiser, F., Schwarz, H., Jakob, M.: Using Wikipedia-based conceptual contexts to calculate document similarity. International Conference on the Digital Society **0** (2009) 322–327
15. Gabrilovich, E., Markovitch, S.: Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In: Proceedings of the 20th International Joint Conference on Artificial Intelligence. (2007) 6–12
16. Anderka, M., Stein, B.: The ESA retrieval model revisited. In: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval, ACM (2009) 670–671
17. Gabrilovich, E., Markovitch, S.: Overcoming the brittleness bottleneck using Wikipedia: Enhancing text categorization with encyclopedic knowledge. In: Proceedings of the Twenty–First National Conference on Artificial Intelligence, American Association for Artificial Intelligence, AAAI Press (2006) 1301–1306
18. Grimm, J.: Berechnung semantischer Ähnlichkeit kleiner Textfragmente mittels Wikipedia. Master thesis, Technische Universitt Darmstadt (Sep 2009)
19. Chakrabarti, S.: Mining the Web: discovering knowledge from hypertext data. Morgan Kaufmann Publishing (2003)
20. Yang, Y.: An evaluation of statistical approaches to text categorization. Information retrieval **1**(1) (1999) 69–90
21. Gurevych, I.: Using the structure of a conceptual network in computing semantic relatedness. In: Proceedings of the 2nd International Joint Conference on Natural Language Processing (IJCNLP 2005), Springer (2005) 767–778