J Intell Inf Syst (2017) 48:309–327
DOI 10.1007/s10844-016-0416-5

Path-based methods on categorical structures for conceptual representation of wikipedia articles

Łukasz Kucharczyk[1] · **Julian Szymański**[1]

Received: 12 January 2015 / Revised: 11 May 2016 / Accepted: 13 May 2016/
Published online: 11 June 2016

Abstract Machine learning algorithms applied to text categorization mostly employ the *Bag of Words* (BoW) representation to describe the content of the documents. This method has been successfully used in many applications, but it is known to have several limitations. One way of improving text representation is usage of Wikipedia as the lexical knowledge base – an approach that has already shown promising results in many research studies. In this paper we propose three path-based measures for computing document relatedness in the conceptual space formed by the hierarchical organization of a *Wikipedia Category Graph* (WCG). We compare the proposed approaches with the standard *Path Length* method to establish the best relatedness measure for the WCG representation. To test overall WCG efficiency, we compare the proposed representations with the BoW method. The evaluation was performed with two different types of clustering algorithms (OPTICS and K-Means), used for categorization of keyword-based search results. The experiments have shown that our approach outperforms the standard Path Length approach, and the WCG representation achieves better results than BoW.

Keywords Text representation · Documents categorization · Information retrieval

1 Introduction

One of the most important tasks during automatic text processing is establishing whether two documents are related to each other, i.e. whether they cover similar topics. To achieve this goal, it is necessary to extract the most significant characteristics for document representation, and then pass them as an input to an appropriate similarity measure.

✉ Julian Szymański
julian.szymanski@eti.pg.gda.pl

[1] Department of Computer Systems Architecture, University of Technology, Faculty of Electronics, Telecommunications and Informatics, ul. Gabriela Narutowicza 11/12, 80-233 Gdańsk, Poland

A widely used approach is based on the *Bag of Words* (*BoW*) method, where each document is represented as a normalized vector of weighted term frequencies (Manning et al. 2009) and then compared, usually using *Cosine Similarity*. This method has a number of known drawbacks; specifically, it ignores relations between terms. It is especially problematic in the case of short documents because they can cover different aspects of the same topic without using any common keywords explicitly.

This shows that the notion of similarity used by BoW, based on simple words co-occurrence, is too narrow to handle more complicated cases that can be solved using more advanced methods. E.g. problems of synonymy and hypernymy can be solved by adding lexical information from an external knowledge base. The final result of such extended representations is computation of *semantic similarity* which is more accurate, but still cannot deal with less obvious relations between words e.g. the conceptual relation between *engine* and *fuel*.

Recent research addresses these problems by enriching or replacing the standard *BoW* features with *conceptual-based* lexical information. This approach focuses on computation of *semantic relatedness* between texts, which covers any kind of lexical and *functional* association which can exist between words.

Some of these approaches successfully employ a Wikipedia Category Graph (*WCG*) as a source of additional text features, although they take relatively little advantage of categories' hierarchical organization (Medelyan et al. 2009). One of the most promising directions of employing categorical structures for abstract document representation is through the usage of WCG hierarchical relations between concepts. Intuitively, the WCG feature space should perform better than the standard BoW, because it has already introduced a basic form of concept classification.

Building such representations is not an easy task, and it requires successful implementation of at least two major steps: a) automatic tagging of arbitrary documents with Wikipedia categories, preferably through deployment of a multi-label classifier; b) extraction of significant relations between categories through appropriate relatedness measures. This paper describes our research, which focuses on the latter of these tasks.

The main contribution of this paper is the introduction of new relatedness measures based on Wikipedia Category Graph. We further extend ideas introduced by (Zesch and Gurevych 2007) who showed that a WCG can be successfully used for computation of semantic relatedness. We first apply their ideas to the concrete application of natural language processing i.e. document clustering. Based on these experiments we identify several drawbacks of the approach proposed by (Zesch and Gurevych 2007) and introduce three possible modifications to overcome them (Section 3.1). In Section 4.1 we show that one of our proposed methods outperforms both baselines established by (Zesch and Gurevych 2007).

For the experiments in our research we use documents which are *a priori* tagged with Wikipedia categories. However, the described approach can be used for any text, if it is tagged with categories, that can be performed: e.g. using a large scale text classifier (Draszawka and Szymański 2013). We expect that the document clustering task should especially benefit from using a representation based on WCG, so we employ the OPTICS and K-Means algorithms for evaluating our approaches.

This paper is an extended version of research shown in (Kucharczyk and Szymański 2014). It proposes three different methods for improving path-based relatedness measures that have been evaluated using two document clustering algorithms. The main changes compared to (Kucharczyk and Szymański 2014) are:

1. Expanded Section 2 describing related works

2. Introduced Section 3 describing our approach in more details
3. New experiments with OPTICS algorithm (which take into account different shapes of *Reachability Plots*)
4. New experiments with K-Means algorithm
5. New experiments for Bag of Words representation

2 Related works

Previous works that employ WCG usually traverse a small part of the category hierarchy, and use relatively simple weighting schemes. Both (Syed et al. 2008) and (Hu et al. 2008) traverse the hierarchy only up to the depth of three levels, while others use only categories directly related to the articles (Banerjee S. et al. 2007; Hu et al. 2009). Other approaches (Gabrilovich and Markovitch 2007) completely resigned from the usage of Wikipedia categories after unsatisfying results obtained with the Open Directory Project, which is also a hierarchical category system (Gabrilovich and Markovitch 2006).

Gabrilovich and Markovitch (2007) argue that hierarchical organization of data violates the orthogonality requirement, which is crucial for computing concept relatedness, and *Explicit Semantic Analysis* is proposed for text representation. These observations are similar to those made by other researchers, who have noticed that limiting WCG search depth improves the results and helps avoiding various anomalies (Strube and Ponzetto 2006; Syed et al. 2008; Hu et al. 2008). Through the depth limit they are able to reduce the size of the processed hierarchy, and thus limit the impact of the orthogonality problem.

On the other hand, it has been shown that WCG shares many important properties with other semantic networks like WordNet (Zesch and Gurevych 2007). In addition (Zesch et al. 2007a) have adopted the classical *relatedness measures*[1] to WCG and concluded, that WCG can be effectively used for natural language processing tasks. (Hamp et al. 1997) also compares the efficiency of WCG with that of the GermaNet and finds out that Wikipedia outperforms GermaNet in regard to computing *semantic relatedness*. Similar experiments performed by (Strube and Ponzetto 2006) on WordNet also confirm that this direction gives promising results.

Moreover, despite the above-mentioned difficulties, researchers who employed limited usage of WCG, reported consistent improvement in their results. Specifically (Banerjee S. et al. 2007; Hu et al. 2008; Hu et al. 2009; Yazdani and Popescu-Belis 2011) adopted Wikipedia categories for enhancing document clustering, while (Sorg and Cimiano 2012; McCrae et al. 2013) used them for cross-lingual and multilingual information retrieval.

Those various controversies regarding WCG show several important directions for future studies. Although (Gabrilovich and Markovitch 2007) did not perform any empirical evaluation of WCG to support their claims (and admit that Wikipedia possesses much less noise than ODP), their observation regarding orthogonality seems valid enough to be taken into consideration. Because the orthogonality problem applies mainly to the standard *cosine* (or *euclidean*) measure, we have decided to analyze more the graph-oriented approach presented by (Zesch et al. 2007b).

[1] *Semantic similarity* is typically defined via the lexical relations of synonymy and hypernymy, while *semantic relatedness* is defined to cover any kind of lexical or functional association that may exist between two words (Budanitsky and Hirst 2006).

Having analysed several relatedness measures, (Zesch and Gurevych 2007; Zesch et al. 2007b) established that *path-based* ones are the most appropriate for WCG. In our research we aim at developing in this direction.

2.1 Explicit semantic analysis algorithm

The basic idea to use a WCG as a representation of text is inspired by *Explicit Semantic Analysis*, where natural language text is mapped to a weighted vector of Wikipedia *articles* (Gabrilovich and Markovitch 2007).

Each Wikipedia Article is considered as a *concept* c_j and represented using BoW weighted with a TF-IDF scheme. Upon this representation, an inverted index is built. A single entry of this index can be interpreted as a vector quantifying the strength of association between a word and concepts.

The last component is a semantic interpreter which uses a constructed index to compute semantic relatedness scores. Given a document d for comparison, the semantic interpreter represents as the TF-IDF vector v, where v_i is the weight for word w_i. For each word w_i in d the matching entry k^i is retrieved from the inverted index, where k^i_j quantifies the strength of association of word w_i with concept c_j. The retrieved concept vector k^i is weighted by the TF-IDF score v_i of word w_i. Then all retrieved concept vectors are aggregated by summing the weights of the corresponding concepts:

$$c_j = \sum_{w_i \in D} v_i \times k^i_j \tag{1}$$

After mapping, the standard *cosine similarity* measure can be used to compute the relatedness of two documents (Fig. 1).

2.2 Graph theoretical analysis, and WordNet measures

Wikipedia is the largest collaboratively created encyclopedia. It is not only available for free, but also offers information of a very good quality (Medelyan et al. 2009).

The knowledge in Wikipedia is not limited to the raw text of encyclopedic entries, but is also encoded in the network structure of Wikipedia pages. This network consists of two components: hyperlink graphs and category graph (WCG). In particular, WCG is organized

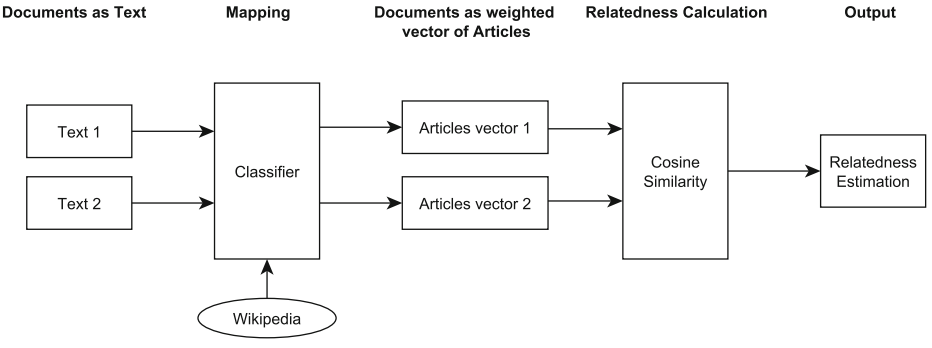

Fig. 1 Schema of computing text relatedness using the ESA approach

as a taxonomy-like structure where categories and subcategories are connected by a *generalization* relation. Moreover, each category can have an arbitrary number of parent categories (and vice versa: each parent category can have any number of subcategories). For example, the category *vehicle* has subcategories like *aircraft* or *watercraft*. Thus, WCG is very similar to semantic word-nets like WordNet or GermaNet.

Zesch and Gurevych (2007) performed a graph-theoretical analysis of the Wikipedia Category Graph to verify whether it shares a common structure with other lexical knowledge bases like WordNet. The conclusions indicate that all such networks have properties of *small world graphs* (i.e. - have large *clustering coefficients* and small *average shortest path length*), and are scale free (i.e. their degree distribution follows the *power law*). Analysis has shown that WordNet and WCG possess highly similar values of graph parameters. These results suggested that WCG could be used for the same NLP applications as WordNet. In particular, measures designed for WordNet should also be applicable to WCG.

Following the above observation (Zesch and Gurevych 2007) examined several WordNet measures to verify the above hypothesis:

- Path Length (PL) between two nodes measured in the number of edges,
- Lowest common subsumer of two nodes,
- Relative corpus frequency.

Performed experiments compared the WCG and GermaNet efficiency in computing semantic relatedness (GermaNet had been used as a baseline). Zesch and Gurevych (2007) used several datasets consisting of word pairs. For each word pair the relatedness value has been assigned by a human judge (*Golden Standard*). Then the same dataset is passed to the particular measures and generated relatedness values are compared with those assigned by human judges. Each measure had to be evaluated twice: once for the WCG version, and once for the GermaNet one. Then, the overall performance of the WCG and GermaNet can be compared.

To perform their experiments (Zesch and Gurevych 2007) first adopted the WordNet measures to the specificity of WCG. The WCG nodes (i.e. Wikipedia categories) do not represent single terms but generalized concepts and thus provide too narrow coverage. To overcome this difficulty, the particular term is maped onto the Wikipedia article with a matching title. This way the task of computing semantic relatedness between *terms* is transformed to the task of computing semantic relatedness between Wikipedia articles.

The conclusions of (Zesch and Gurevych 2007) are that the WCG outperforms the baseline established by GermaNet in computing semantic relatedness. They also observed that among the examined types of relatedness measures, the path-based ones turned out to be the most successful.

3 Proposed methods

In our approach we propose combining the methods presented by (Gabrilovich and Markovitch 2007) and (Zesch and Gurevych 2007). The process shown in Fig. 2 uses Wikipedia *categories* (instead of *articles*), and then exploits their hierarchical organization by employing *Path Length* measures for computing documents' relatedness (instead of *cosine similarity*). To evaluate the proposed approach we can remove the classifier component and provide a set of *a priori* tagged documents instead. The general idea is presented in Fig. 3. It is the easiest way to obtain such a testing set is to use actual Wikipedia articles. Such an experimental setting has two benefits:

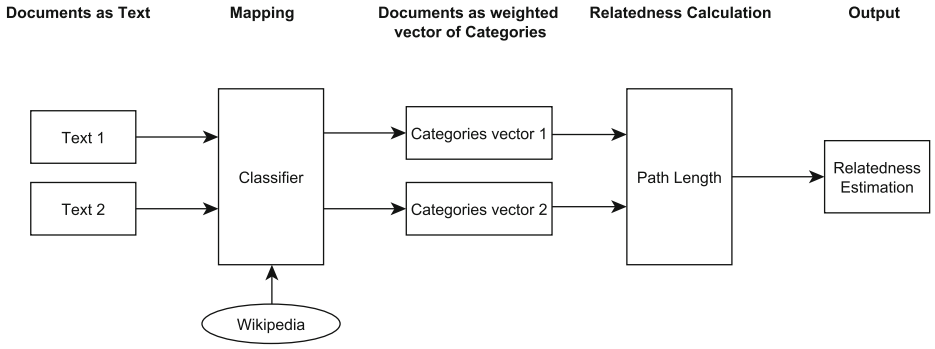

Fig. 2 Schema of computing text relatedness employing WCG

– Usage of *a priori* tagged documents allows us to avoid additional noise, which could be introduced to the experiment by a classifier. This way we can establish an upper boundary for the efficiency of our proposed methods (assuming the human–made category assignments are correct).
– We can first test the basic validity of our assumptions before engaging in the time-consuming task of building a good, large-scale text classifier, like (Fan et al. 2008; Xue et al. 2008).

By employing the proposed representation for documents clustering we can achieve better results than the standard Bag of Words method (Fig. 4). Moreover, we propose several methods of computing document relatedness in such a conceptual space. The experiments have shown they perform better than baseline measures derived from WordNet.

While comparing our approach to (Zesch and Gurevych 2007) it is important to point out the most significant differences:

– Zesch and Gurevych (2007) computed relatedness between word pairs to show that WCG can be applied to NLP. We actually apply WCG to a concrete NLP task i.e. document clustering (which also *involves* computation of semantic relatedness). Thus our research is a logical continuation of those performed by (Zesch and Gurevych 2007),
– Because our preliminary experiments turned out to give unsatisfactory results (see Sections 6 and 7 for the results obtained with measures proposed by (Zesch and Gurevych 2007)) we developed several modifications to these measures (see Section 4).

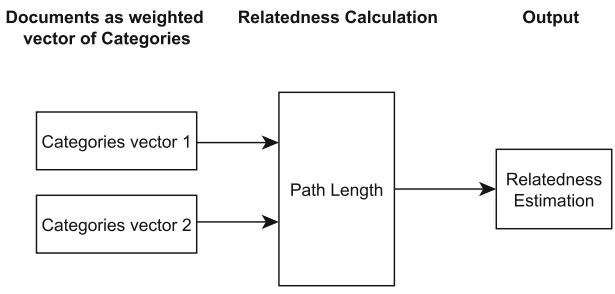

Fig. 3 Schema of computing simplified relatedness based on WCG

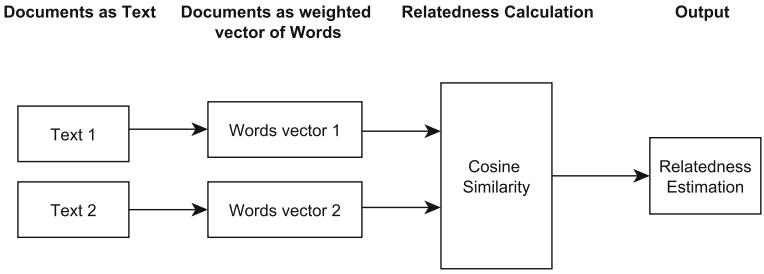

Fig. 4 Schema of computing text relatedness using Bag of Words

We chose document clustering for the evaluation of the proposed methods as this task should benefit from exploitation of non-classical relations. It is worth to note that our test setting was similar to the one used by (Zesch and Gurevych 2007) i.e. we use Wikipedia articles directly for evaluation purposes without performing any additional mapping between words/documents and Wikipedia articles.

4 Path-based relatedness measures

In order to perform clustering of documents represented as WCG concepts, we need to establish a method for computing semantic relatedness. Path-based techniques define relatedness as the distance between nodes in a concept graph (Zesch and Gurevych 2007). In Section 4.1, we present the baseline method, and describe observations we made using it for document clustering. Then, in Sections 4.2–4.4 we present three methods that we offered in (Kucharczyk and Szymański 2014), which were developed to overcome the baseline.

4.1 Baseline method: path length (PL)

Path Length is one of the standard WordNet-based relatedness measures adapted for WCG by (Zesch et al. 2007b; Zesch and Gurevych 2007), which has also been shown to be the most successful. Thus, we employ this method as a baseline for comparison to approaches proposed by us. The measure is defined as a path length between two nodes (number of edges along the shortest path):

$$dist_{PL}(c_i, c_j) = length(c_i, c_j) \tag{2}$$

As each Wikipedia article can be assigned to multiple categories, different methods have been proposed for their aggregation (Zesch and Gurevych 2007). Having defined C_1 and C_2 as the set of categories assigned to the articles a_1 and a_2, respectively. Then the PL distance is computed for each category pair (c_k, c_l) with $c_k \in C_1$ and $c_l \in C_2$. Then, we use either the minimum value or the average over all computed pairs:

$$dist_{PL+Min}(a_1, a_2) = \min_{c_i \in C_1, c_j \in C_2} dist_{PL}(c_i, c_j) \tag{3}$$

$$dist_{PL+Avg}(a_1, a_2) = \frac{\sum_{c_i \in C_1} \sum_{c_j \in C_2} dist_{PL}(c_i, c_j)}{|C_1||C_2|} \tag{4}$$

Both adaptation schemes possess a serious drawback, which make them unsuitable for document clustering. $PL + Min$ ignores all information provided by categories placed

outside the shortest path. This way a lot of information about documents similarity (or dissimilarity) is lost.

Let us consider three documents a_1, a_2, a_3 and their assigned categories $C_1 = \{History, Technology\}, C_2 = \{History, Technology\}, C_3 = \{History, Art\}$. In such a setting $PL + Min$ would give the result:

$$dist_{PL+Min}(a_1, a_2) = dist_{PL+Min}(a_1, a_3) = dist_{PL+Min}(a_2, a_3)$$

instead of:

$$dist_{PL+Min}(a_1, a_2) < dist_{PL+Min}(a_1, a_3) = dist_{PL+Min}(a_2, a_3)$$

If the clustered document set contains many articles sharing a common category, $PL + Min$ will assign the same distance value to all of them, effectively preventing the construction of meaningful clusters.

On the other hand, $PL + Avg$ suffers from a quite opposite weakness. Averaging over all possible category pairs gives excessive weight to redundant or unusual ones, which is especially problematic if documents possess different numbers of categories.

Let us consider two documents a_1, a_2 and their assigned categories $C_1 = \{Technology, Engines, Hi-Tech\ Industry, American\ Company\}$, $C_2 = \{American\ Company\}$. Clearly, both documents should be clustered together (assuming there are no other technical-oriented articles), however, their distance value will be artificially increased by the fact that categories in C_1 are mostly redundant.

So, each of these adaptation schemes causes problems when applied to the task of document clustering. They either discard too much information (PL+Min) or give excessive weight to redundant categories and noise (PL+Avg). In consequence, both schemes become sensitive to data distribution within the clustered collection, which makes extraction of clusters difficult.

4.2 Method 1: semi-average path length (PL+Avg*)

In this method we overcome the weakness of *Path Length* by introducing a new adaptation scheme. Instead of calculating average over all category pairs we apply the following procedure:

For given documents a_1, a_2 and their assigned categories C_1, C_2 we calculate distances between each category and its *opposite document* i.e. for each category in C_1 we calculate its distance to document a_2, and for each category in C_2 we calculate distance to a_1. Then we calculate the average over all these distances:

$$dist_{PL+Avg*}(a_1, a_2) = \frac{\sum_{c_i \in C_1} dist_{PL}(c_i, a_2) + \sum_{c_j \in C_2} dist_{PL}(c_j, a_1)}{|C_1| + |C_2|} \quad (5)$$

The major difference between this method and $PL + Avg$ is that in this setting we compute distances between category and article, so only the several most sensible category pairs are calculated. This way, redundant categories do not impose much penalty on the final distance score.

4.3 Method 2: Semi-average path length with frequency reduction (PL+Avg*+DF)

This is an extension of the previous method $PL+Avg*$, where we additionally ignore nodes with *Document Frequency* values (Manning et al. 2009) below the given threshold. This

way we remove unnecessary noise in the same way as using the *Bag of Words* representation (Liu et al. 2003).

During the WCG traversal, if a node is found below the defined document frequency threshold, an $infinite$ distance is assigned to it (it is effectively treated as unreachable). However, its descendants are processed normally, with the only difference that the cost of travel through the reduced node is equal to zero. As a result, all descendants of the removed node are considered as being one level closer to its source document:

$$dist_{PL+DF}(a, c) = dist_{PL}(a, c) - |R_{a,c}| \tag{6}$$

$$R_{a,c} = \{r : r \in p_{a,c} \wedge DF(r) < THRESHOLD\} \tag{7}$$

where $p_{a,c} = (a, ..., ..., c)$ is the sequence of nodes lying on the path between article a and category c, and $R_{a,c}$ is a subsequence of $p_{a,c}$ consisting of those nodes whose document frequency is below $DF(c) < THRESHOLD$.

Then, we compute $dist_{PL+Avg*+DF}$ in the same way as defined in Section 4.2 by substituting the values of $dist_{PL}$ in (5) with the values of $dist_{PL+DF}$.

4.4 Method 3: Minimum weighted path length (PL+Min+IDF)

This method is similar to the baseline $PL+Min$ by using only the shortest path for distance calculation. However $PL + Min + IDF$ does not use the simple *Breadth First Search* approach, but instead of it, performs a shortest path search with the *Dijkstra* algorithm. In this method WCG is interpreted as a *weighted* graph where travel cost through node c_i (where c_i is a category) is equal to the inverse of its *IDF* statistic (Manning et al. 2009):

$$IDF(c_i) = log \frac{N}{DF(c_i)} \tag{8}$$

$$cost(c_i) = \frac{1}{IDF(c_i)} \tag{9}$$

where N is the number of documents in clustered collection. This way, travel through common terms should be more expensive and we expect it to generate greater differences in documents distance values.

5 Evaluation procedure

For evaluation of the methods presented in Section 4, we have used Wikipedia articles as *a priori* tagged document set. In our future research, we are going to employ a multi-label classifier to automatically assign Wikipedia categories (Draszawka and Szymański 2013), so clustering of any documents could be performed. At this stage, we use Wikipedia articles for testing purposes, as the usage of a classifier could introduce additional noise to experiments (see Section 2.1).

To make our evaluation procedure closer to real world applications, we have implemented a clustering search engine. The user interface of our system is shown in Fig. 5. It allows us to group keyword-based search results within Wikipedia in a similar fashion as

Fig. 5 User interface of WikiClusterSearch engine

Clusty[2] or Carrot[3] does for WebPages. For document clustering we used two different types of algorithms:

OPTICS – a hierarchical clustering algorithm which generates a tree-like structure to describe relations between documents. In the case of most hierarchical algorithms (particularly various versions of HAC) this structure is usually visualized as a *dendogram*. Because dendograms are difficult to interpret for humans, we used the OPTICS algorithm which generates a more human-readable cluster hierarchy, so its results are easier to evaluate and analyze.

K-MEANS as one of the most popular partitioning clustering algorithms is commonly employed as a referential algorithm in many research studies (Jain 2010; Steinbach et al. 2000). This algorithm divides input data into disjoint classes, where each document is

[2] http://clusty.com/

[3] http://search.carrot2.org

assigned to exactly one class. It is widely used by industry and the scientific community due to its simplicity.

Using these fundamentally different clustering algorithms for evaluation, we intend to ensure that received results are not algorithm-specific, and to show that our approach for text representation can be (potentially) generalized for different data-mining applications.

For the evaluation of the proposed methods we have decided to use *external validation measures*, which are considered to be more accurate than *internal* ones (Draszawka and Szymański 2011). For this purpose we have manually prepared sets of reference groupings - one set for hierarchical partitioning and one set for flat partitioning using the following procedure:

1. Keyword search within Wikipedia has been performed to select a set of articles containing a specified phrase
2. Returned results have been manually grouped into clusters. That structure was then saved as reference partitioning $C^T = \{C_1^T, ..., C_{K^T}^T\}$ (we denote C_k^T sets as *classes*, K^T is the number of classes). We call such manually prepared partitioning C^T as a *Gold Standard* (Manning et al. 2009). We prepared two Gold Standards for each set of documents: one for evaluation using hierarchical clustering (i.e. OPTICS algorithm) and one for evaluation using flat clustering (i.e. K-Means algorithm).
3. Returned results have been used again as an input for automatic clustering, using methods described in Section 4. Calculated output was then saved as partitioning $C^C = \{C_1^C, ..., C_{K^C}^C\}$ for comparison (we call C_i^C sets as *clusters*, K^C is the number of clusters).
4. Partitions C^T and C^C have been compared to calculate an evaluation score.

Due to the limited human resources, we were able to prepare nine Gold Standards (each in two variants: flat and hierarchical), each consisting of approximately 100 documents. Building such a testing set is a time-consuming task. Even assuming that assignment of each document to an appropriate cluster requires only one minute, then creation of single Gold Standard would require at least one and a half hours of work. In practice however, creation of single hierarchical Gold Standard took approximately four hours.

6 Evaluation with OPTICS algorithm

We have used $PL + Min$ and $PL + Avg$ (described in Section 4.1) as the two baselines for the evaluation, both of which have already been reported to provide good results for the computation of category relatedness (Zesch and Gurevych 2007; Zesch et al. 2007b). We have expected that the modifications which we introduced to those measures should remove the problems mentioned in Section 4.1 and make them suitable for document clustering. Tests performed on several prepared Gold Standards have shown that one of our proposed methods, $PL + Avg*$, has achieved better results than both baselines (Table 1).

We also used the BoW representation with *cosine similarity* to evaluate overlay efficiency of WCG with *path-based* measures and their suitability for document clustering. Because we use *a priori* tagged documents as input data, we expect to receive considerably better results than those of BoW. In such an experimental setting there should be no errors caused by invalid category assignments, so the obtained results can only be attributed to the inherent properties of BoW or WCG.

Table 1 Method evaluation: hPMCC

	BoW	PL+Min	PL+Avg	PL+Avg*	PL+Avg* +DF	PL+Min +IDF
Class	0.0	0.43	0.42	0.56	0.47	0.52
Game	0.0	0.31	0.25	0.38	0.48	0.15
Jaguar	0.74	0.7	0.73	0.73	0.56	0.57
Kernel	0.0	0.31	0.5	0.57	0.68	0.42
Relation	0.0	0.65	0.38	0.45	0.43	0.1
Sphere	0.0	0.24	0.36	0.52	0.33	0.35
Element	0.0	0.29	0.39	0.52	0.43	0.22
Feature	0.0	0.15	0.29	0.42	0.25	0.19
Part	0.0	0.41	0.35	0.6	0.33	0.34
Average	0.08	0.39	0.41	0.53	0.44	0.31

Hierarchical clustering with the OPTICS algorithm ($MinPts = 3, \epsilon = infinity$, RATIO $= 0.75$)

6.1 Experimental setting

OPTICS is a density-based, hierarchical clustering algorithm designed to identify clusters of varying density (Ankerst et al. 1999). Its unique feature is the ability to exploit density information to build a specific *ordering* of data points. This ordering can either be visualized as a *reachability plot* (e.g. Fig. 6) or passed to an extraction algorithm to produce a hierarchy of clusters.

Because the original method of extracting clusters from *reachability plots* is sensitive to input parameters (Sander et al. 2003) (and thus inappropriate for evaluation purposes),

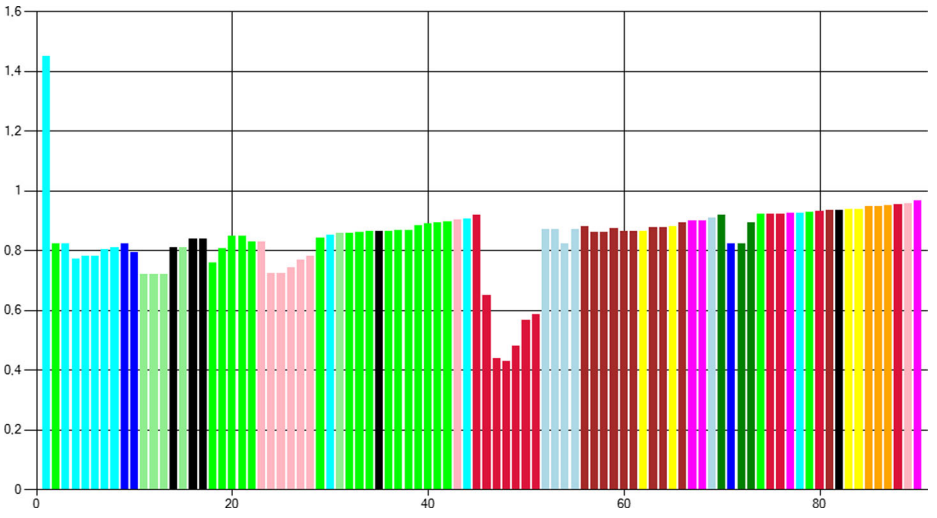

Fig. 6 Example of shallow reachability plot with proper document ordering

Table 2 Method evaluation: hPMCC

	BoW	PL+Min	PL+Avg	PL+Avg*	PL+Avg* +DF	PL+Min +IDF
Class	0.38	0.46	0.49	0.56	0.52	0.52
Game	0.19	0.31	0.31	0.4	0.48	0.23
Jaguar	0.74	0.77	0.75	0.75	0.61	0.75
Kernel	0.31	0.32	0.75	0.61	0.68	0.47
Relation	0.52	0.62	0.43	0.45	0.46	0.59
Sphere	0.21	0.34	0.4	0.52	0.34	0.41
Element	0.27	0.39	0.4	0.55	0.41	0.3
Feature	0.19	0.2	0.29	0.42	0.27	0.22
Part	0.42	0.54	0.54	0.6	0.54	0.44
Average	0.36	0.44	0.48	0.54	0.48	0.44

Hierarchical clustering with OPTICS algorithm ($MinPts = 3$, $\epsilon = infinity$, RATIO $= argmax(Pmcc(x))$)

we have used the *Cluster Tree*[4] , algorithm which is *almost* parameterless – its single input argument, the *ratio of significance*, is strongly recommended to be set to a fixed value of 0.75 (Sander et al. 2003). However, during our experiments we have observed that the final extraction results can vary significantly depending on the exact values of this parameter. For example, the reachability plot shown in Fig. 6 correctly groups different document classes, i.e. bars of the same color are grouped together. However, application of the default extraction parameters would yield a final hierarchy of poor quality. It is due to the fact that the examined plot is very shallow.

We were not able to establish a global optimal value for the extraction parameters – any fixed value tended to favor some plot shapes and penalize others. It turned out to be especially visible in the case of the Bag of Words representation, which produces very shallow plots.

As the goal of our research is not to analyze different extraction schemes for the OPTICS algorithm, we decided to adjust our evaluation procedure accordingly. To minimize the impact of the extraction algorithm on the final evaluation score, we use the whole range of possible parameter values and then select the best result for each method (*ratio of significant separation* in range from 0.0 to 1.0 with the step equal to 0.05 i.e. RATIO $\in \{x : x \in (0, 1) \wedge k \in N, x = 0.05k\}$). The results are shown in Table 2. For the completeness of evaluation we also present the results for a default parameter value i.e RATIO $= 0.75$ (Table 1).

For the evaluation of hierarchical clustering we have employed the hierarchical version of *Pmcc* measure called *hPmcc* (Draszawka and Szymański 2011)[5]. For each partitioning,

[4]OPTICS does not create a hierarchy of clusters but so called *reachability plots*. To obtain hierarchical clustering it is necessary to employ an additional algorithm e.g. *Cluster Tree*

[5]*hPmcc* is a modification of the standard *Pmcc* measure which can be used for the evaluation of hierarchical clusters.

C^T and C^C (see Section 5), we calculate a NxN *non-binary similarity matrix* $\mathbf{S}_{ca}$:

$$\mathbf{S}_{ca} = [s_{ca}(i, j)], \quad s_{ca}(i, j) = \frac{lev(NCC, Root)}{(mean(lev(i, Root), lev(j, Root))}$$

where NCC is the abbreviation of the *Nearest Common Cluster* and *lev* function denotes the level-based distance between clusters in which objects i and j are found (Draszawka and Szymański 2011). Then, using the *similarity matrices* $\mathbf{S}^T_{ca}$ and $\mathbf{S}^C_{ca}$, we calculate *hPmcc* using the same formula as for the standard *Pmcc* measure (Draszawka and Szymański 2011):

$$Pmcc(\mathbf{S}^T, \mathbf{S}^C) = \frac{1}{(M-1)\sigma^T\sigma^C} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} (s^T_{i,j} - \mu^T)(s^C_{i,j} - \mu^C) \tag{10}$$

6.2 OPTICS results

The tests performed on the prepared Gold Standards have shown that one of our proposed methods, $PL + Avg*$, has achieved significantly better results than both baselines (Tables 1 and 2).

It can also be seen that $PL + Min$ introduces the highest variance of the results, just as expected. The analysis of generated Reachability Plots has shown, that this method generates very well formed plots if the data distribution is favorable (as can be seen for the *Relation* Gold Standard), although it generally fails to capture significant inter-document differences and thus extracts less clusters. $PL + Avg$ also behaves as expected – its Reachability Plots are significantly flattened, due to averaging of the distance values.

Surprisingly, the $PL + Min + IDF$ method gains a very low score (especially for the default $ratio$ value, Table 1). We assumed that using the collection level information would allow us to improve the results, but instead we received Reachability Plots with very shallow dents. High density of WCG turned out to provide many alternative routes for the traversal algorithm, so the introduction of additional weights has made the distance score approach its average value. $PL + Avg* + DF$ have also performed below our expectations. We have assumed that removal of the apparently noisy nodes from WCG should improve the clustering results. The better scores in the *Game* Gold Standard partially confirm this assumption, but unfortunately this solution turned out to be not globally optimal. The deleted nodes often proved to hold additional information about distances, and their removal had generally a negative impact on the performance of the otherwise successful method $PL + Avg*$.

The first observation regarding the *BoW* representation is the fact that it fails to extract *any* clusters for the default parameter value of $ratio = 0.75$ (as already mentioned in Section 6.1). The only exception is a very good score for *Jaguar* Gold Standard. The WCG representation turned out to be less dependent on the exact value of $ratio$, especially the $PL + Avg*$ method. Its maximum efficiency (Table 2) is very close to that achieved for the default parameter value (Table 1).

The second observation is that the *Jaguar* Gold Standard is the only one that got successfully extracted for default $ratio = 0.75$ (Table 1). Moreover, it achieved very good scores both for the default and optimal $ratio$ values (Tables 1 and 2). The reason for this behavior is that *Jaguar* can be considered the 'easiest' Gold Standard from the testing set. Nearly all methods got their highest scores for *Jaguar* (except for $PL + Avg* + DF$, that best score is for *Kernel*). It is due to the fact that its underlying documents set form natural hard clustering (i.e. clusters do not overlap). This effect is additionally strengthened by the fact that for

Table 3 Method evaluation: F-Measure

	BoW	PL+Min	PL+Avg	PL+Avg*	PL+Avg* +DF	PL+Min +IDF
Class	0.61	0.72	0.73	0.74	0.71	0.75
Game	0.66	0.73	0.67	0.75	0.76	0.63
Jaguar	0.79	0.86	0.76	0.86	0.83	0.78
Kernel	0.75	0.84	0.83	0.88	0.82	0.75
Relation	0.82	0.84	0.87	0.88	0.85	0.86
Sphere	0.57	0.79	0.72	0.74	0.82	0.69
Element	0.52	0.71	0.69	0.69	0.74	0.72
Feature	0.53	0.69	0.65	0.69	0.65	0.6
Part	0.67	0.73	0.64	0.75	0.69	0.68
Average	0.66	0.77	0.73	0.78	0.76	0.72

Best result (i.e. the best seed) listed in the table. Flat clustering with K-Means algorithm. 50 iterations with different seeds

each cluster in *Jaguar* Gold Standard, there exist several characteristic words, which clearly distinguish a given cluster from the others e.g. *car*, *panther*, *album*, *fender*.

The other interesting issue is why did BoW achieve *better* scores than $PL + Avg*$ for *Relation* Gold Standard, and why did BoW perform *worse* than $PL + Min$ at the same time (Table 2). The main reason turns out to be the variable density of WCG. Some categories have many subcategories, whilst others have only a few[6]. Moreover, those branches can vary greatly in their total depth. We can observe that $PL + Min$ performs best in such cases. Because it considers only categories belonging to the shortest path, it ignores much of the noise emerging from variable graph density.

7 Evaluation with K-Means algorithm

To extend the the evaluation of the WCG representation performed with the OPTICS algorithm, we additionally conducted another experiment with K-means. The primary goal of using two different algorithms is to verify whether the results obtained in Section 6.2 are independent of any specific algorithm used. Analogically, for the test performed in Section 6.2 we use $PL + Min$ and $PL + Avg$ as two baselines for the path-based similarity measures, and BoW as a reference for the WCG representation in general.

7.1 Experimental setting

Because of the random initialization of the K-Means algorithm (Manning et al. 2009) we cannot base our analysis on its single run, but we need to rely on aggregated statistics from its multiple runs. Thus, for each Gold Standard, we executed the K-Means algorithm for 50 different initial seeds. Then all generated partitions were evaluated using F-Measure to find the best and average scores (Tables 3 and 4 respectively).

[6]The same is true the other way i.e. some categories have multiple parents while others have only a few.

Table 4 Method evaluation: F-Measure

	BoW	PL+Min	PL+Avg	PL+Avg*	PL+Avg* +DF	PL+Min +IDF
Class	0.49	0.59	0.61	0.61	0.6	0.62
Game	0.5	0.56	0.57	0.65	0.63	0.51
Jaguar	0.55	0.67	0.57	0.59	0.61	0.53
Kernel	0.65	0.65	0.66	0.66	0.65	0.57
Relation	0.66	0.7	0.72	0.72	0.7	0.71
Sphere	0.66	0.63	0.58	0.62	0.66	0.57
Element	0.5	0.6	0.6	0.61	0.63	0.59
Feature	0.45	0.5	0.51	0.54	0.54	0.51
Part	0.46	0.58	0.56	0.59	0.56	0.53
Average	0.53	0.61	0.6	0.62	0.62	0.57

Average result over 50 executions listed in table. Flat clustering with K-Means algorithm. 50 iterations with different seeds

The other important characteristic of K-Means, which must be taken into account, is the fact that it requires specification of *number of clusters* as an input parameter (Manning et al. 2009). To overcome this difficulty we pass the *expected* number of clusters from a particular Gold Standard. This way we do not have to introduce any additional procedure to estimate the optimal number of clusters.

It should be noted that absolute numeric scores obtained for K-Means cannot be directly compared with those received for OPTICS (presented in Section 6.2). Although we use the same document sets, there are important differences. First, we had to prepare different Gold Standards for each algorithm (hierarchical for OPTICS and flat for K-Means). Second, we also had to use different evaluation measures for each algorithm (hPmcc for OPTICS and F-Measure for K-Means). In consequence, the comparisons of K-Means and Optics can be performed only in a qualitative manner rather than a quantitative one.

7.2 K-means results

For the evaluation of flat clustering we used a standard *F-measure* which is the weighted harmonic mean of precision and recall (Manning et al. 2009) described by Formula (11).

$$F = (\beta^2 + 1)\frac{precision \cdot recall}{\beta^2 \cdot precision + recall} \tag{11}$$

We used the standard value of $\beta = 1$ which gives equal weight to precision and recall described by formula (12).

$$F = \frac{2 \cdot precision \cdot recall}{precision + recall} \tag{12}$$

Although exact evaluation scores for OPTICS (Tables 1, 2) and K-Means (Tables 3, 4) cannot be directly compared, it is worth noting that overlay behavior of considered measures is quite consistent for these algorithms. The WCG representation with Path-Based measures gets considerably better results than the standard BoW representation with Cosine Similarity. Particularly, for both K-Means and OPTICS algorithms, the *Element*, *Game* and *Sphere* Gold Standards achieve the most significant improvement, whilst *Jaguar* and *Relation* get the smallest.

Moreover, for both K-Means and OPTICS algorithms, $PL + Avg*$ measure gets the best average performance, although in the case of K-Means the differences between particular Path Length methods turn out to be quite small (in contrast to OPTICS described in Section 6.2). It is caused by the fact that the structure of flat Gold Standards is by definition simpler than hierarchical ones. In consequence, it is less crucial for flat clustering to efficiently exploit hierarchical information embedded in WCG.

The achieved results show that the WCG representation performs better than BoW, and that among examined *path-based* measures $PL + Avg*$ achieves the best performance.

8 Conclusions and future directions

In this paper we proposed a method of using the Wikipedia categories as an alternative document representation for clustering. We introduced three path-based methods for document relatedness in the conceptual space formed from WCG. To demonstrate our approach in a real-life application we have implemented a clustering search engine and used it to perform empirical evaluation of the proposed methods. Our experiments have shown that one of the proposed measures, $PL + Avg*$, achieves better scores than both baseline methods, and that the proposed representation based on WCG performs better than the BoW approach.

In our further research we plan to use a *large scale multi-label text classifier* (Draszawka and Szymański 2013) for automatically tagging raw text with Wikipedia categories. This should allow us to apply the WCG representation for a wider scale rather than only for pretagged Wikipedia articles. Once a classifier is integrated into the system, it would become possible to use the standard benchmark collections, like *Reuters*, for system evaluation. Due to the fact that we have put a lot of effort into constructing hierarchical Gold Standards we plan to make them available on-line and enhance them with existing text processing benchmarks.

Users expect a near real-time response from search engines, and this expectation imposes considerable performance requirements on the clustering algorithms. To improve performance of our system (described in Section 5) we plan to integrate additional optimizations into the underlying OPTICS algorithm. It can be achieved by implementing indexes similar to those described in (Kryszkiewicz and Lasek 2010).

The performed tests have shown that the WCG representation achieves better scores than BoW, and also that there is still potential for further improvement. The most serious problem caused by WCG is the fact that some of its parts have a very variable semantic density – some hierarchical paths contain very specific concepts while others only general ones. If a method for semantic analysis, such that it takes into account specificity of a category, is developed, then the efficiency of the representation based on WCG could be further increased.

The proposed method of representation based on conceptual space can also be used for domain-oriented repositories that offer a category system. One of the potential applications of our approach is adaptation of the presented clustering search engine to the *MEDLINE* repository. This way, the methods proposed in this paper could significantly improve searching for documents within the medical domain.

Acknowledgments This research was partially funded by grants from National Centre for Research and Development (PBS2/A3/17/2013, Internet platform for data integration and collaboration of medical research teams for the stroke treatment centers).

References

Ankerst, M., Breunig, M.M., Kriegel, H., & Sander, J. (1999). Optics: Ordering points to identify the clustering structure., *ACM Sigmod record*, (Vol. 28 pp. 49–60): ACM press.

Banerjee S., Ramanathan K., & Gupta A. (2007). Clustering short texts using wikipedia. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 787–788): ACM. SIGIR '07.

Budanitsky, A., & Hirst, G. (2006). Evaluating WordNet-based measures of lexical semantic relatedness. *Computational Linguistics*, *32*, 13–47.

Draszawka, K., & Szymański, J. (2011). External validation measures for nested clustering of text documents. In *ISMIS Industrial Session* (pp. 207–225).

Draszawka, K., & Szymański, J. (2013). Thresholding strategies for large scale multi-label text classifier. In *2013 The 6th International Conference on Human System Interaction (HSI)* (pp. 350–355).

Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., & Lin, C.-J. (2008). Liblinear: a library for large linear classification. *The Journal of Machine Learning Research*, *9*, 1871–1874.

Gabrilovich, E., & Markovitch, S. (2006). Overcoming the brittleness bottleneck using wikipedia: Enhancing text categorization with encyclopedic knowledge. In *Proceedings of the 21st National Conference on Artificial Intelligence - Volume 2* (pp. 1301–1306): AAAI Press. AAAI'06.

Gabrilovich, E., & Markovitch, S. (2007). Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *IJCAI*, (Vol. 7 pp. 1606-1611).

Hamp, B., Feldweg, H., & et al. (1997). GermaNet-a lexical-semantic net for german. In *Proceedings of ACL workshop Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications, Citeseer* (pp. 9–15).

Hu, J., Fang, L., Cao, Y., Zeng, H.J., Li, H., Yang, Q., & Chen, Z. (2008). Enhancing text clustering by leveraging Wikipedia semantics. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 179–186). New York: ACM. SIGIR '08.

Hu, X., Zhang, X., Lu, C., Park, E.K., & Zhou, X. (2009). Exploiting wikipedia as external knowledge for document clustering. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 389–396): ACM.

Jain, A.K. (2010). Data clustering: 50 years beyond k-means. *Pattern Recognition Letters*, *31*(8), 651–666.

Kryszkiewicz, M., & Lasek, P. (2010). Ti-dbscan: Clustering with dbscan by means of the triangle inequality. In *Rough Sets and Current Trends in Computing* (pp. 60–69): Springer.

Kucharczyk, Ł., & Szymański, J. (2014). Evaluation of path based methods for conceptual representation of the text. In *Proceedings of the Foundations of Intelligent Systems - 21st International Symposium, ISMIS 2014, Roskilde, Denmark, June 25–27, 2014* (pp. 435–444).

Liu, T., Liu, S., & Chen, Z. (2003). An evaluation on feature selection for text clustering. In *In ICML* (pp. 488–495).

Manning, C.D., Raghavan, P., & Schtze, H. (2009). *An introduction to information retrieval*. Cambridge: Cambridge University Press.

McCrae, J.P., Cimiano, P., & Klinger, R. (2013). Orthonormal explicit topic analysis for cross-lingual document matching. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Seattle* (pp. 1732–1740).

Medelyan, O., Milne, D., Legg, C., & Witten, I.H. (2009). Mining meaning from wikipedia. *International Journal of Human Computer Studies*, *67*(9), 716–754.

Sander, J., Qin, X., Lu, Z., Niu, N., & Kovarsky, A. (2003). Automatic extraction of clusters from hierarchical clustering representations. In *Advances in Knowledge Discovery and Data Mining* (pp. 75–87): Springer.

Sorg, P., & Cimiano, P. (2012). Exploiting Wikipedia for cross-lingual and multilingual information retrieval. *Data & Knowledge Engineering*, *74*, 26–45.

Steinbach, M., Karypis, G., Kumar, V., & et al. (2000). A comparison of document clustering techniques., *KDD Workshop on text mining, boston*, (Vol. 400 pp. 525-526).

Strube, M., & Ponzetto, S.P. (2006). Wikirelate! computing semantic relatedness using wikipedia. In *Proceedings of the 21st national conference on Artificial intelligence* (pp. 1419–1424): AAAI Press.

Syed, Z.S., Finin, T., & Joshi, A. (2008). Wikipedia as an ontology for describing documents. In *ICWSM*.

Xue, G.-R., Xing, D., Yang, Q., & Yu, Y. (2008). Deep classification in large-scale text hierarchies. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 619–626): ACM.

Yazdani, M., & Popescu-Belis, A. (2011). Using a wikipedia-based semantic relatedness measure for document clustering. In *Proceedings of TextGraphs-6: Graph-based Methods for Natural Language Processing, Association for Computational Linguistics, Stroudsburg, TextGraphs-6* (pp. 29–36).

Zesch, T., & Gurevych, I. (2007). Analysis of the Wikipedia Category Graph for NLP Applications. In *Proceedings of the TextGraphs-2 Workshop (NAACL-HLT)*.

Zesch, T., Gurevych, I., & Mühlhäuser, M. (2007a). Analyzing and accessing wikipedia as a lexical semantic resource. *Data Structures for Linguistic Resources and Applications*, 197–205.

Zesch, T., Gurevych, I., & Mühlhäuser, M. (2007b). Comparing Wikipedia and German WordNet by Evaluating Semantic Relatedness on Multiple Datasets. In *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*.