Semantic Relatedness Metric for Wikipedia Concepts Based on Link Analysis and its Application to Word Sense Disambiguation

© Denis Turdakov, Pavel Velikhov

ISP RAS
turdakov@ispras.ru, pvelikhov@yahoo.com

Abstract

Wikipedia has grown into a high quality up-to-date knowledge base and can enable many knowledge-based applications, which rely on semantic information. One of the most general and quite powerful semantic tools is a measure of semantic relatedness between concepts. Moreover, the ability to efficiently produce a list of ranked similar concepts for a given concept is very important for a wide range of applications. We propose to use a simple measure of similarity between Wikipedia concepts, based on Dice's measure, and provide very efficient heuristic methods to compute top k ranking results. Furthermore, since our heuristics are based on statistical properties of scale-free networks, we show that these heuristics are applicable to other complex ontologies. Finally, in order to evaluate the measure, we have used it to solve the problem of word-sense disambiguation. Our approach to word sense disambiguation is based solely on the similarity measure and produces results with high accuracy.

1 Introduction

Wikipedia is the leading open encyclopedia that has evolved into a comprehensive resource with very good coverage on diverse topics, important entities, events, etc. The English Wikipedia currently contains over 4 million articles (including redirection articles). Furthermore, Wikipedia contains quite a bit of structured information: it has a rich category structure, separate pages for ambiguous terms, and structured data for certain types of articles. Finally, it contains over 90 million links between articles. Most of these links signify that some semantic relationship holds between the source and target concepts and can used to compute a measure of relatedness that typically outperforms traditional text similarity measures.

In our work we have used the links between Wikipedia concepts to compute a semantic relatedness measure[1]. While there has been a number of prior works that introduced a variety of similar measures, we present a simple measure that yields good results and at the same time can be computed efficiently. Specifically, we address the problems of computing top k similar articles, given a specific Wikipedia article and computing the similarity measure between two articles. We present simple yet powerful heuristics that yield orders of magnitude performance improvements over the straightforward ranking method. The heuristics are also applicable to other complex ontologies, since they are based on statistical properties of complex networks. In prior work some authors have evaluated their measure directly using human ranking. Instead, we have validated our approach by solving the problem of word sense disambiguation, for which a number of reference points is available. We believe such comparison is more reliable and easier to conduct, than setting up human experiments.

The rest of the paper is organized as follows. In Section 2 we present out measure of semantic relatedness and discuss its use in a number of applications. In Section 3 we raise the issue of efficient computation of relatedness measures and provide a set of heuristics that significantly improve the performance of ranking with our measure and computing the relatedness between a pair of concepts. In Section 4 we generalize our results to other complex ontologies. Then, in Section 5 we describe our approach in solving the problem of word sense disambiguation using the relatedness measure and demonstrate our results. Finally, we conclude with future work in Section 6.

2 Semantic Relatedness

Wikipedia has recently been widely recognized as an enabling knowledge base for a variety of intelligent systems. However, Wikipedia is not suitable for

Proceedings of the Spring Young Researcher's Colloquium On Database and Information Systems SYRCoDIS, St.-Petersburg, Russia, 2008

[1] Sometimes semantic similarity is used interchangeably with semantic relatedness, however we consider the later term to be more specific.

machine consumption as is and in order to bootstrap various tools with Wikipedia knowledge some semantic extraction needs to take place. In our work we extract a simple relatedness measure between Wikipedia articles that proves useful in a variety of tasks such as query refinement in search engines, document classification, document clustering, faceted browsing, etc. Traditionally, the above applications either use basic vector-space similarity functions (in case of classification and clustering [13]) or rely on pre-built ontologies (for query refinement and faceted browsing[14]). With a high quality semantic relatedness measure, we can greatly enhance the quality of results in these applications. For example, [12] presents a new classification technique that uses a semantic relatedness measure and achieves excellent results.

2.1 Related Work

We can divide the previous work in similarity measures into two broad classes: basic measures inspired by traditional IR metrics, such as the cosine metric [3,4, 11], and graph theoretic measures, such as SimRank[8,12]. While the second class typically provides a better quality measure, the computational efficiency of these methods is not high enough to be used in practical data intensive applications. Therefore, in our work we analyze a measure based on Dice's measure that is commonly used in IR.

2.2 Weighted Dice Metric

Dice's measure has a very intuitive meaning: in case of Wikipedia articles two pages will be related, if the fraction of the links they have in common to the total number of links of both pages is high. More formally,

$$Dice(A,B) = \frac{2 \times |n(A) \bigcap n(B)|}{|n(A)| + |n(B)|}$$

where *n(A)* is the set of articles linking (considering both incoming and outgoing links) to article *A*, and *n(B)* is the set of articles linking to *B*. While exploring the structure of Wikipedia, we have noticed that some types of links are extremely relevant to semantic relatedness, while some other types lead to wrong results. Hence we have added a weighting scheme to the basic measure, based on the following link types:

See Also Links: Most Wikipedia articles have a See Also section that lists related articles. These links explicitly signify that a linked page is semantically related. Therefore, see also links are very important for semantic relatedness and we assign the highest weight to the links of this type. Inverse See Also Links (incoming links) are also quite important and they receive a high weight also.

Double links: Articles that link to each other directly by regular links in most cases turn out to be quite related, hence these types of links come next in our weighting scheme.

Links between articles in the same category: Wikipedia has a rich category structure, and articles belonging to the same category are related. However, some categories are very broad and consist of unrelated articles. For example, the category *"Indian Actors"* contains over 600 articles and the degree of relatedness between all of these actors is not very significant. Therefore, we identify articles that have both: a link between them, and share the same category, as the next most relevant type of link.

The rest of the links are just regular Wikipedia links, but we separate out of them into further types: Date links and Template Links, which receive the lowest weights in our scheme. In our experiments we have used the following weighting scheme, shown in Table 1.

See Also	5	Double Link:	2
Inverse See Also	2	Common Category	1.5
Regular Link	1	Inverse Regular Link	0.5
Date	0.1	Template	0.1

Table 1: Weights for various link types

Finally, we have also incorporated IDF weighting into our measure. Since our measure is based on Dice (and not cosine), we don't scale the IDF with a logarithmic function, we simply use it as a multiplier. While the IDF-enhanced measure produces better subjective results, it does not significantly improve our evaluation results on Word Sense Disambiguation problem.

3 Efficient Ranking and Computation

In many applications, such as document classification, faceted browsing and word sense disambiguation, we need to retrieve the list of articles related to a given article, ranked by their relatedness measure. Also, some applications might need to precompute relatedness scores in order to speed up online processing. Typically, only a small percentage of top ranking related concepts are needed in these examples. It turns out that with the scale of Wikipedia, the task of ranking top k articles for a query becomes non-trivial. For example, the article *"United Kingdom"* has over 80 thousand incoming and outgoing links and thus has non-zero relatedness score with over a million other Wikipedia articles. A naïve approach towards computing the top k results for such articles becomes intractable. Therefore, in order to build practical systems that make use of the relatedness measure, we have to provide heuristics that may sacrifice the quality of the measure slightly, but yield significant improvements in computational efficiency. Below we present four heuristics that dramatically limit the search space of potential related concepts. All of these heuristics are based on the observation that related articles typically link to each other or have a common link with another, highly related article.

3.1 Limiting the Search Space

OL Heuristic (Outgoing links). Our first heuristic is targeted at articles with large degrees. When we are computing top k related pages for these articles, we can limit our search only to the outgoing links, which are a

tiny percentage of incoming links for articles with a high degree. For example, *"United Kingdom"* has only 900 outgoing links, although its total degree is over 80 thousand. However, this heuristic works well for very large articles, but produces poor results on articles with intermediate degrees.

OL Top 20 Heuristic. We improve the previous heuristic by making another observation: similar articles that don't have a direct link between them, will link to a closely related article. Our second heuristic expands the search for top k related articles by including all regular outgoing links of the top 20[2] ranked articles, produced by the OL heuristic. This still leads to very efficient raking, since the average number of outgoing links in Wikipedia is about 90 and grows slowly with the total number of links.[3]

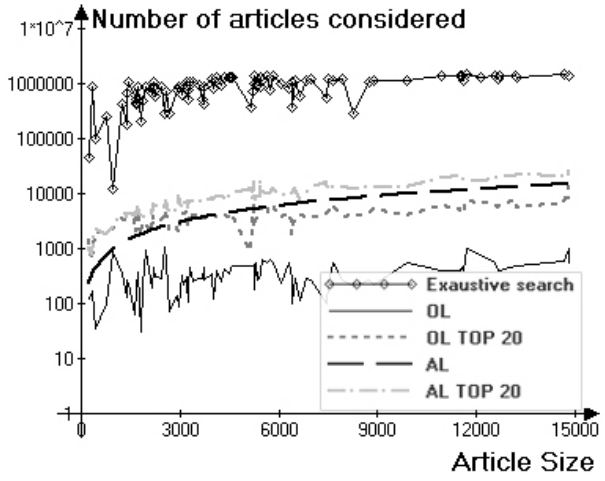

Figure 1: Performance of heuristic methods

AL (All Links) and AL Top 20 Heuristics. For articles with a relatively small number of links, the first two heuristics yield poor results. However, since the degrees of the articles are smaller, we can expand the search space without too much computational penalty. AL heuristic considers all articles with an incoming or an outgoing link. AL Top 20 expands the search space of AL with regular outgoing links of 20 top ranking articles.

3.1 Efficient Computation of Relatedness Measure

So far we have focused on limiting the search space of potential high-ranking articles. However, computing the similarity measure is also an expensive operation, since Wikipedia articles have thousands of links. To address this problem, we propose to use a simple randomized algorithm to compute an approximate similarity measure.

[2] We choose to pick 20 top ranking articles, since it gives us a good tradeoff between improved accuracy and slightly higher computational complexity.

[3] The number of incoming links, on the other hand, grows fast and reaches very high numbers, as in the case of *"United Kingdom"*.

AL Random Sample Heuristic. This heuristic picks 100 links randomly from a pair of articles and computes the relatedness based on this sample. It performs surprisingly well, producing a ranking very similar to the AL heuristic.

3.2 Accuracy and Performance

The accuracy and performance of the above heuristics are presented in Figure 1 and 2. We demonstrate the accuracy of out methods by comparing the top 20 ranked results produced by the heuristic method with the exhaustive search. To estimate the efficiency, we plot the search space of our heuristics vs. the search space of an exhaustive method. Due to the computational difficulties of performing exhaustive search that is required in the comparisons, we used a sample of articles that have at most 15000 links. However, the trends can be easily extrapolated to larger articles.

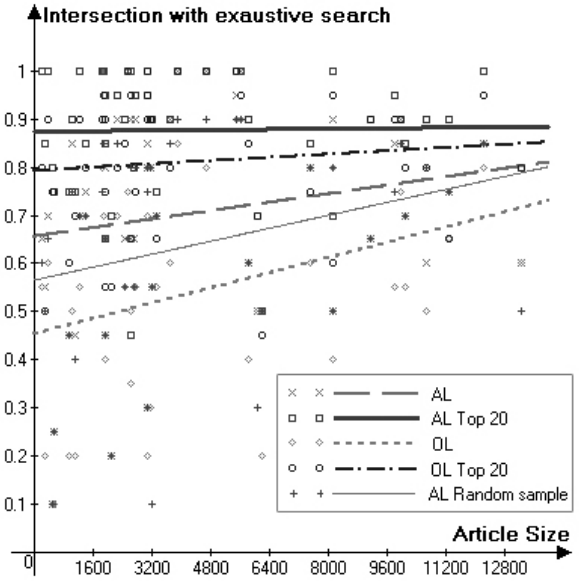

Figure 2: Accuracy of heuristic methods

4 Other complex ontologies

As we have mentioned above, our results are applicable not only to Wikipedia, but also to other graphs with similar properties. Wikipedia is an instance of scale-free network[9], that has two important properties: the degree distribution of nodes (articles) follows a power law, and the clustering coefficient (the probability that two articles have a direct link, if they are connected through a chain of length two) is higher than in random graphs and also follows a power law. These properties were discovered for Wikipedia in [10,13]. We give an informal proof for our heuristics, based on the clustering property of Wikipedia articles. Given an article a, its top k related articles will have a large number of links in common with a, which is especially true for articles with a large in and out degrees. In such case, the probability that there will be a

direct link from article a to an article within top k related articles is very high due to the clustering property. Our measurements for article of similar sizes yield a clustering coefficient of $7*10^{-3}$. Now, lets consider articles that are related to *"United Kingdom": "Labour Party"* is in the 20th rank wrt. to our relatedness measure. *"United Kingdom"* and *"Labour Party"* have about 3000 common neighbours, hence the probability that there will be a direct link between these articles is extremely close to 1. For articles with such a high degree, the OL heuristic is sufficient to achieve near 100% accuracy. As we consider articles with a smaller degrees, the clustering coefficients increase, however the number of common neighbours for top ranking articles drops very quickly. Therefore, we need to employ heuristics that expand the search space beyond first degree neighbours.

5 Word Sense Disambiguation

Word Sense Disambiguation (WSD) is a complicated but extremely important task in natural language processing. Unresolved ambiguous words can for example significantly decrease the precision of text classification: the word "platform" can be used in the expression "railway platform", or it can refer to a hardware architecture or a software platform. Without disambiguating the meaning of ambiguous words, text classification, information retrieval and other algorithms will produce erroneous results.

While a lot of research has been carried out on this topic, the problem is still considered hard. For us WSD represents a perfect test bed to evaluate our relatedness measure: we use a very simple method, based solely on semantic relatedness, and compare our results with existing approaches.

In the next subsection we will take a look at some of WSD methods that utilize Wikipedia and compare them with our approach.

5.1 Related work

There are two basic approaches to disambiguation: approach based on machine learning [1,2] and methods based similarity models [3,4,5,6]. [1] investigates the method for enhancing performance of supervised learning algorithms. One significant drawback of supervised systems they are applicable to those few words for which sense tagged data is available, and their accuracy is strongly connected to the amount of labeled data available at hand. The approach described in [1] uses Wikipedia as a source of sense annotations for building sense tagged corpora. Authors suggested to use Wikipedia link structure for generation of sense annotated corpora. We use a similar method to generate the test corpora for evaluating our method.

A similar method was presented in [2]. The authors employed Wikipedia entity pages, redirection pages, categories, and hyperlinks and built a context-article cosine similarity model and an SVM based on a taxonomy kernel. They evaluated their models for person name disambiguation over 110, 540, and 2,847 categories, reporting accuracies between 55.4% and 84.8% on (55-word context, entity) pairs extracted from Wikipedia, depending on the model and the development/test data employed.

The method that is most similar to our approach is presented in [3]. The authors use Explicit Semantic Analysis (ESA), a novel method that represents the meaning of texts in a high-dimensional space of concepts derived from Wikipedia. ESA works by first building an inverted index from words to all Wikipedia articles that contain them. Then, it estimates a relatedness score for any two documents by using the inverted index to build a vector over Wikipedia articles for each document and by computing the cosine similarity between the two vectors.

5.2 Computing candidate word meanings

We make use of Wikipedia's disambiguation pages and redirection articles obtain candidate meanings of ambiguous words. Then we use our relatedness measure to pick the meaning that has the highest relevance to the context where the ambiguous word appeared. Wikipedia contains special types of articles for ambiguous terms. For each ambiguous term these pages contain all of the word's meanings, which are separate articles in Wikipedia with their own link structure. For example the article "platform (disambiguation)" contains 16 meanings of the word "platform". At the end of 2007 there were more then 80000 disambiguation pages in Wikipedia and this number is growing.

Wikipedia disambiguation pages are not completely structured and there are no simple rules that guarantee an error-prone extraction of all possible meanings. We illustrate this problem with an example. Usually different meanings are listed in the first paragraph of the disambiguation page or in lines marked with "*". But often such lines have links to pages that are unrelated to the ambiguous term. For example the article *"war (disambiguation)"* have the following choice among its meanings:

"War", a song by Joe Satriani off "*The Extremist*" [4]

If we collect the links from such lines we will have quite a few erroneous disambiguation candidates. Instead, we pick only the first link from each line of the disambiguation page if the text preceding the link doesn't contain the ambiguous term itself or its acronym. (We skip the link in the example above because it has the term "war" appearing in the text before the link).

Some ambiguous terms that stem from case sensitivity of Wikipedia don't have corresponding disambiguation pages, but we can still infer the different meanings of the terms. We convert all Wikipedia terms to upper case and create disambiguation pages when we have conflicts. For example, *"SUX"* and *"Sux"* point to *"Sioux Gateway Airport" and "Suxamethonium chloride"* correspondingly, and we create the appropriate disambiguation page for them.

[4] The links are shown in italic font

For our experiments we select 7 closest terms from the context where the ambiguous term appears, then compute the semantic distance by two methods: a naïve method, described in the next subsection only uses the Wikipedia graph, and the semantic relatedness measure that we introduced above.

5.3 WSD Method based on Wikipedia graph (Naïve)

For the naïve method, we define semantic distance between the context and a candidate term meaning as the amount of common articles in the neighborhoods[5] of the context and the term in Wikipedia.

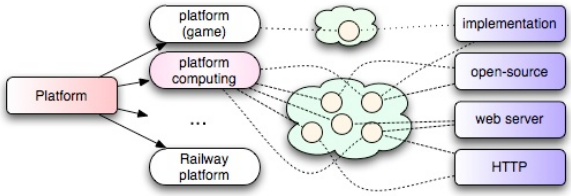

Figure 3: Naive WSD Method

For example, consider the disambiguation process for the term "platform" in the text *"Jigsaw is W3C's open-source project that started in May 1996. It is a web server platform that provides a sample HTTP 1.1 implementation and ... "*. There are four Wikipedia concepts around this word: *"open-source"*, *"web server"*, *"HTTP"* and *"implementation"*. For each of the 16 meanings of the word *"platform"*, the system calculates the amount of common articles between the neighborhood of each meaning and the union of the neighborhoods of each topic from the context. The biggest intersection is with the neighborhood of the *"platform (computing)"* article. It has 122 common articles. Next candidate is *"platform game"* with only 18 common articles. Therefore, we decide that in this context the meaning of the word *"platform"* is a computer platform.

This simple method shows good precision (62.2%), as illustrated in Table 1, however it has some drawbacks. First, it uses only nearest neighbors to compute semantic distance. But sometimes semantically related articles will not have a direct link between each other. However, if we take into account all second neighbors of article, the WSD task will became too computationally complex. Next problem is that links between two articles can be semantically poor. For example, a link between *"Moscow"* and *"Fahrenheit"* makes little sense. This can significantly decrease the precision of this method. Hence, a natural way to increase precision of this technique is to use the relatedness metric in order to compute the similarity between a candidate meaning and its context.

[5] By a neighbor of an article we define all Wikipedia articles that have an incoming or an outgoing link to the original article.

5.4 WSD based on Semantic Relatedness

We improve over the naïve method by using the measure of semantic relatedness. Given a meaning candidate c, and a set of context terms $t_1 \ldots t_n$, we compute a ranked list of related articles $R(c)$ for c and $R(t_i)$ for each t_i in $t_1 \ldots t_n$. We use the heuristic AL, described in the previous section, in order to obtain these lists efficiently. The distance between c and $t_1 \ldots t_n$ is computed as follows. We first take a union of $R(t_i)$ lists, summing up weights of repeating articles, and obtain $R(T)$. Then we compute the similarity between $R(c)$ and $R(T)$ using the following methods: Sum of products, Cosine, Dice and Jaccard measures. We apply these four measures in the naïve approach as well, where the weights are binary. Finally, we pick the candidate that maximizes the given similarity function.

5.5 Experiments and Results

One of the problems of evaluating WSD techniques is that there are no conventional benchmarks. Also, WSD problem has many variations and must be carefully scoped. For instance most NLP tools solve the problem of Part-of-Speech tagging, which is a special case of WSD, however we only focus on disambiguating proper nouns. Hence, we created our own benchmarks from the Wikipedia content itself. Wikipedia links often have the following structure:

[[*part1* | *part2*]],

where *part1* is a normalized Wikipedia topic and *part2* is text in the link anchor that is presented to the user. If *part2* is an ambiguous term, then *part1* is usually a Wikipedia topic corresponding to one of the term meanings. We use the dictionary created from disambiguation pages to parse Wikipedia articles and find links with ambiguous terms in *part2*, which at the same time have one of disambiguation candidates in *part1*. For example, there is a link [[*platformism* | *platform*]] in the article *"Anarchism"*. Here it means that ambiguous term *"platform"* has a meaning *"platformism"*.

	Naive	Semantic Relatedness
Intersection	**61,83**	71,8
Cosine	59,82	72,52
Dice	61,42	**72,58**
Jaccard	60,28	**72,58**

Table 2: Comparison of WSD methods

Therefore, we don't we don't rely on NLP methods to search for ambiguous words and we can easily get the correct answer for the disambiguation using *part1*. We used this technique for constructing a small test corpus with 1000 ambiguous terms in order to evaluate our methods.

The best result (72,58%) is obtained by using the Dice and Jaccard measures with the semantic relatedness method. This results shows that we achieve much better precision when using the semantic

relatedness measure in comparison with the naïve approach (best result 61,83%). The results are summarized in Table 2.

We also ranked potential word meaning by similarity weight in descending order and inspected the top 2 and 3 results. The average number of meanings for an ambiguous word was 17,65 for our test set. Almost 87% of right answers were in top two positions of sorted lists and top 3 positions contained more then 91% of right answers. This result is summarized in Table 3.

The difference between similarity scores of the right and first answers was very small. Hence, there is a potential to improve our results further, by improving our context modeling methods and incorporating linguistic analysis. Currently, some verbs that are homonyms to nouns are frequently detected as nouns. For example, the verb "*aims*" was detected as acronym "*AIMS*". Furthermore, due to the limitations of our text parser, we detect Wikipedia terms only in normal form. We believe that by improving our method to avoid these problems we will achieve the performance close to that of human experts.

	Top-2	Top-3
Intersection	85,12	89,03
Cosine	87.46	90.86
Dice	**86.95**	**91.38**
Jaccard	**86.95**	**91.38**

Table 3: Percentage of top ranked correct meanings

5 Conclusions and Future Work

We have presented a simple measure of semantic relatedness, based on the link structure of Wikipedia. We addressed the problem of computing this measure efficiently and have provided heuristics for computing top k related articles. These heuristics achieve high accuracy, but limit the search space drastically and make the approach suitable for practical use in a variety of data intensive systems. We also presented a randomized algorithm to compute the relatedness measure between two articles efficiently and shown that its accuracy in ranking is very close to the true measure. In order to evaluate the quality of the measure, we have presented a simple method for word sense disambiguation, based on the relatedness measure. We evaluated our approach and found it to perform on par with the competing approaches and close to the performance of human experts.

In future work, we will explore more heuristics with the aim to produce a single tunable method with an explicit analysis of computational complexity and graceful degradation. Furthremore, we plan to formally prove the quality of the heuristics using the statistical properties of scale-free networks. This will enable us to estimate the quality of a specific tunable heuristic in advance, without the need to experiment. Finally, we are planning to explore more sophisticated methods of modelling context, similar to the method presented in [12] and computing term-context similarity measure. We also plan to incorporate linguistic analysis into our text parser and and we expect to improve WSD results significantly.

References

[1] Rada Mihalcea. Using Wikipedia for AutomaticWord Sense Disambiguation. Proceedings of NAACL HLT 2007, pages 196–203, Rochester, NY, April 2007

[2] Razvan Bunescu, Marius Pasca. Using Encyclopedic Knowledge for Named Entity Disambiguation.

[3] Gabrilovich, E. and S. Markovitch. 2007. Computing semantic relatedness using Wikipedia-based explicit semantic analysis. *Proceedings of IJCAI*, 1606-1611

[4] Strube, M. and S. P. Ponzeto. 2006. WikiRelate! Computing semantic relatedness using Wikipedia. In *Proceedings of AAAI*, 1419-1424.

[5] Silviu Cucerzan. Large-Scale Named Entity Disambiguation Based on Wikipedia Data. In *Proc. 2007 Joint Conference on EMNLP and CNLL*, pages 708–716, Prague, The Czech Republic, 2007.

[6] Simon Overell, Joao Magalhaes and Stefan Ruger. Place disambiguation with co-occurrence models

[7] Mihalcea, R., T. Chklovski, and A. Kilgarriff. The Senseval-3 English lexical sample task. In *Proceedings of SENSEVAL-3*, 25-28

[8] Glen Jeh , Jennifer Widom. SimRank: a measure of structural-context similarity. Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, July 23-26, 2002, Edmonton, Alberta, Canada

[9] Albert R. and Barabási A.-L. Statistical mechanics of complex networks. Rev. Mod. Phys. 74, 47–97 (2002).

[10] Turdakov D. Recommender system based on user-generated content. Proceedings of the Spring Young Researcher's Colloquium on Database and Information Systems SYRCoDIS, Moscow, Russia, 2007

[11] David Milne, Computing Semantic Relatedness using Wikipedia Link Structure, Procroceedings of New Zealand Computer Science Research Student Conference NZCSRSC, 2007

[12] Ollivier Y. and Senellart P. Finding Related Pages Using Green Measures: An Illustration with Wikipedia, In Proceedings of the 22nd National Conference on Artificial Intelligence (AAAI'07), Vancouver, Canada, 22-26 July 2007

[13] Voβ, J. Measuring Wikipedia. Proceedings of 10th International Conference of the International Society for Scientometrics and Informetrics, (Stockholm, Sweden), 2005.

[14] Maciej Janik, Krys Kochut. "Wikipedia in action: Ontological Knowledge in Text Categorization", UGA Technical Report No. UGA-CS-TR-07-001, November 2007