Proceedings of the Federated Conference on
Computer Science and Information Systems pp. 241–248
ISBN 978-83-60810-51-4

Improving the Wikipedia Miner Word Sense Disambiguation Algorithm

Aleksander Pohl
Jagiellonian University
ul. Łojasiewicza 4, 30-348 Kraków, Poland
Email: aleksander.pohl@uj.edu.pl

***Abstract*—This document describes the improvements of the Wikipedia Miner word sense disambiguation algorithm. The original algorithm performs very well in detecting key terms in documents and disambiguating them against Wikipedia articles. By replacing the original Normalized Google Distance inspired measure with Jaccard coefficient inspired measure and taking into account additional features, the disambiguation algorithm was improved by 8 percentage points (F_1-measure), without impeding its performance nor introducing any additional preprocessing overhead. This document also presents some statistical data that are extracted from the Polish Wikipedia by Wikipedia Miner. An automatic evaluation of the performance of the disambiguation algorithm for Polish shows that it is almost as good as for English, even though the Polish Wikipedia has only a quarter of the number of the articles of the English Wikipedia.**

I. Introduction

WIKIPEDIA Miner [1] is an open source software for mining Wikipedia, developed by David Milne and Ian H. Witten. It is designed as a toolkit that simplifies access to the semantic content of Wikipedia and offers features such as word sense similarity measure and topic detection in documents. The second feature is provided by implementing sense disambiguation of the terms found in a given document against Wikipedia articles. Although Wikipedia is not a traditional semantic dictionary such as WordNet [2] and it does not capture sense of adjectives, verbs and adverbs well, it might be transformed into a vast knowledge base, that works very well as a reference resource covering millions of physical and abstract objects (cf. [3] and [4]). As such it serves very well for providing semantics for nouns and multi-word nominal expressions.

The Wikipedia Miner's ability to disambiguate terms is based on the measure of semantic relatedness [5] that utilizes Wikipeida's internal link structure in assessing the relatedness of the articles and the associated concepts. This measure takes into account the number of incoming links that are and are not common for the articles in question. These values are transformed into the measure using equation inspired by the Normalized Google Distance [6].

The disambiguation decisions made by the algorithm are driven by a decision tree induced from examples taken from Wikipedia [7] using the C4.5 algorithm [8]. The features that are taken into account cover semantic relatedness, sense probability and „goodness" of the context, that is the value indicating if the disambiguation context is consistent.

Although the results obtained by Milne and Witten in providing explanation for the concepts found in textual documents are quite good, there is still some space for improvement. Substituting the original semantic relatedness measure with a different one and taking into account more features it is possible to obtain better disambiguation results.

The document is structured as follows: first, the other word sense disambiguation algorithms that use Wikipedia as a primary knowledge source are discussed. Then the disambiguation algorithm improvements implemented by the author are shortly presented. The next section discusses the link structure of Wikipedia that is crucial for the algorithm, with examples taken from the Polish Wikipedia. This is followed by a detailed description of the disambiguation algorithm, its improvements and the evaluation methodology. The paper is concluded with results obtained both for the English and the Polish Wikipedia.

II. Related work

The work on Wikipedia-based word sense disambiguation algorithms was started by Mihalcea and Csomai in the Wikify! project [9]. They implemented two algorithms: a knowledge-based algorithm and a data-driven algorithm. The first one was based on the Lesk algorithm [10] and used the Wikipedia articles as the definitions of the associated concepts. The second one used local and topical features of the text that were integrated into a machine learning classifier [11]. The examples used to train the classifier were taken from the Wikipedia links. The data-driven algorithm was performing better that the knowledge-based one, achieving performance of 87,73% (F_1-measure).

To some extent the algorithm of Milne and Witten was built upon the results of that work, especially the link probability measure (cf. IV-D) and the conception of using the Wikipedia links as training and testing examples for disambiguation were reused. However the results of Milne and Witten were reported by the authors to be substantially better.

The DBpedia Spotlight project is a recent attempt that tackles the same problem. The design of the system together with the performance comparison of many such systems is presented in [12]. The primary difference between DBpedia

This work is partially sponsored by the Faculty of Management and Social Communication of the Jagiellonian University.

Spotlight and Wikipeida Miner is that the first system may be configured with many parameters, e.g. some specific classes out of the 272 defined in the DBpedia ontology [13] might be selected to identify only these entities that belong to those classes. The second, more important difference, is that the disambiguation algorithm, similarly to the data-driven algorithm of Mihalcea and Csomai, uses the textual context of the links as the primary means for disambiguation. This step requires language-specific preprocessing, since for each link occurrence the surrounding paragraph has to be tokenized, stopworded and stemmed.

To perform disambiguation DBpedia Spotlight does not employ any explicit semantic similarity measure between the concepts. It employs the well known Vector Space Model with the $tf \cdot idf$ measure replaced by $tf \cdot icf$, where c in icf stands for *candidate (resource)*. It is defined by the following equation:

$$ICF(w_j) = log\frac{|R_t|}{n(w_j)} \quad (1)$$

where:

- w_j is the word that has to be weighted
- R_t stands for the set of candidate Wikipedia articles for the term t
- $n(w_j)$ denotes the number of the articles that contain the word w_j; as such it captures the relative importance of the word among the candidate articles

The authors of DBpedia Spotlight concluded that it outperforms six other currently available systems, including Wikipedia Miner. Such a result is indeed very good, but there are several questions that have to be answered, before we can fully embrace that claim. First question concerns the number of texts and disambiguated terms that were used to test these systems. In the testing set there were only 10 short texts containing 165 words on average and 251 concepts in total. So we should ask if such a small amount of data is enough for a comparison of 7 complex systems.

The second question concerns the accuracy reported by the authors: it is claimed that it was 80,5% [12, Table 1], which stays in a large contrast with the results obtained by the author for the Wikipedia Miner algorithm. The accuracy of the original Wikipedia Miner algorithm was above 97% due to the fact that most decisions made by the algorithm concern false negatives. And the last, but not the least is the fact that the performance of the Wikipedia Miner (referred to as M&W system by the authors of DBpedia Spotlight) is not fully reported (cf. [12, Table 2]).

It seems that a more systematic and neutral comparison of the systems should be performed.

III. Approach

The work presented in this article is largely based on the work of Milne and Witten ([7], [5], [1]) and should be seen as its extension. The data that are used for the disambiguation as well as the algorithm structure are the same. There are only two areas of algorithm that were improved: the semantic relatedness measure and the set of features used to induce the decision tree and disambiguate the terms.

The original algorithm uses Normalized Google Distance to measure the semantic relatedness between the Wikipedia articles. By substituting that measure with a Jaccard coefficient inspired measure, the author obtained disambiguation results that were substantially better than the original ones.

The selection of features is another area of the algorithm that was improved. In each disambiguation case we are dealing with a selection of one sense out of n, but the n and the sense probability distribution is different for the different terms. As a result the absolute values obtained for the different terms might be incomparable. Even though two senses of two different terms are most probable, the probability of the first one might be 90% (in case it is dominating) and the second 40% (in case there are several popular senses).

By extending the set of features with relatedness rank and sense probability rank, the author obtained further improvement of the disambiguation results. It should be noted that these new features are trivial to compute and does not require any additional preprocessing overhead.

IV. Link structure

The internal link structure of Wikipedia is central to the Wikipedia Miner disambiguation algorithm. The links are used to recognize the candidate senses for the terms, they are used to measure the semantic relatedness of the senses, they are used to estimate the probability of senses for an ambiguous term and finally an important feature *link probability*, first described in [9] is used to weight the recognized terms. This section describes the various features of the links, illustrating them with examples taken from the Polish Wikipedia.

A. Links as article names

Table I shows an example of the links in the Polish Wikipedia that are used to link to the article about *Poland*. It contains the morphological forms of the word (*Polsce*, *Polski*, *Polska*, ...), as well as the adjectival forms derived from the noun (*polski*, *polska*, *polskiego*, *polskiej*, ...), the forms of the name of Polish citizens (*Polak*, *Polaków*, *Polacy*) and the abbreviation of Poland – *RP*.

The names of the links are the primary means for detecting the candidate senses for the terms to be disambiguated. This approach was first used in the work of Mihalcea and Csomai [9]. It should be noted that the links are not transformed in any way – they are not case folded, nor stemmed, nor lemmatized. This seems to be strange at the first glance, since regarding Polish, which is an inflectional language, stemming or lemmatization seems to be indispensable. But this is not as strange when Table I is considered – it contains almost all[1] the inflectional variants of *Poland* as well as many other variants. What is more – the number of occurrences for each variant clearly indicates which forms are more common and which could be ignored as occasional. Such data are not available

[1]The vocative form *Polsko* is not present in the table, since it has only 13 occurrences.

TABLE I
LINKS REFERRING TO *Poland* IN THE POLISH WIKIPEDIA. (THE LINKS WITH LESS THAN 100 OCCURRENCES ARE NOT SHOWN.)

Link name	# of occurrences
Polsce	74196
polski	11564
Polski	5528
Polska	3271
polska	2234
Polskę	763
Polską	647
polskiego	605
polskiej	510
polsko	458
polskich	432
polskie	295
polskim	246
polską	130
Polak	108
RP	101
Polaków	100
Polacy	100

TABLE II
THE 15 ARTICLES (AND *Poland*) MOST RELATED TO *Warszawa* THAT LINK TO OR ARE LINKED FROM THIS ARTICLE ACCORDING TO sr_G MEASURE. (NOTE: log_2 IS USED AS THE log FUNCTION)

Wikipedia article	sr_G
Uniwersytet Warszawski	0.5460
Kraków	0.5350
Powstanie warszawskie	0.5310
Łódź	0.5099
Poznań	0.4853
Lublin	0.4757
Wrocław	0.4632
Order Virtuti Militari	0.4603
Ulica Okopowa w Warszawie	0.4585
Cmentarz żydowski w Warszawie (Wola)	0.4568
Lwów	0.4525
Gdańsk	0.4479
Politechnika Warszawska	0.4415
Wola (dzielnica Warszawy)	0.4401
Białystok	0.4337
...	
Polska (*Poland*)	0.2201

when one sticks to the canonical title of the article and its redirect pages.

B. Relatedness measure

One of the key features of the Wikipedia Miner toolkit, exploited in the algorithm, is the measure of semantic relatedness based on the links, first described in [5]. The idea behind the measure is simple – it is assumed that the more often the links referring to the articles share their context, the more the articles are related to each other. In this respect this idea is similar to latent semantic analysis [14]. However the actual measure used in the original algorithm [7] is based on the Normalized Google Distance (NGD) [6]. The rationale for this measure and its performance in comparison to the other Wikipedia based measures of semantic relatedness is given in [5]. The original measure looks as follows:

$$sr_G(a,b) = 1 - \frac{log(max(|A|,|B|)) - log(|A \cap B|)}{log(|W|) - log(min(|A|,|B|))} \quad (2)$$

Where:

- $sr_G(a,b)$ – the measure of semantic relatedness between a and b
- $|A|$ – the size of the set of articles that link to a
- $|A \cap B|$ – the size of the set of articles that link both to a and b
- $|W|$ – the number of all articles in Wikipedia

It should be noted that this measure might give values lower than 0, so the implementation constraints the results to non-negative values.

Analyzing the results of the application of this measure to Polish Wikipedia, the author has observed that the measure-based ranking of related articles is strange. The results for the Polish word *Warszawa* (*Warsaw* – the capital of Poland) are given in Table II. The first place is occupied by the *Warsaw University*, there are several Polish cities (*Kraków*, *Łódź*, *Poznań*, ...) and *Warsaw upraising – Powstanie Warszawskie*. What is strange here is the high position of *Ulica Okopowa w Warszawie*, one of the Warsaw streets, definitely not the most known and *Cmentarz żydowski w Warszawie* one of the Jewish cemeteries in Warsaw, definitely not the most standing out place in Warsaw. But what is much more strange is the absence of *Poland* in these results. Are Poland and Warsaw not related?

Comparing the individual values that appear in equation 2, it turned out that if favors articles with only a few links over articles with many links. What is even more strange – the relatedness value might be above 0, even if the articles does not have any linking article in common.

That observation gave impulse to experiment with the other measure that seems to be better suited for the task – Jaccard coefficient [15]. The Jaccard inspired measure is as follows:

$$sr_J(a,b) = \begin{cases} \frac{1}{1-log\left(\frac{|A \cap B|}{|A \cup B|}\right)} & |A \cap B| > 0 \\ 0 & a \neq b \wedge |A \cap B| = 0 \\ 1 & a = b \wedge |A \cap B| = 0 \end{cases} \quad (3)$$

Where:

- $|A \cup B|$ – the size of the set of articles that link to a or b

Table III shows the results of applying this measure to the same task of ranking the articles that link to or are linked from the article about *Warsaw*. First of all the position of *Poland* is much better, but it might be observed that this ranking is also a little strange. There are three numbers there – 1945, 2007 and 2006, which represent years. Their high position stems from the fact that contrary to sr_G this measure favors articles with larger link base. However it is easy to filter them from the results with a simple regular expression, since years with their fuzzy semantics, does not seem to contribute much to the disambiguation problem.

Still the improvement observed for *Warsaw* was obviously not decisive for the general improvement of the measure. It only indicated that changing the measure might be a step in the right direction.

TABLE III
THE 15 ARTICLES MOST RELATED TO *Warszawa* THAT LINK TO OR ARE LINKED FROM THIS ARTICLE ACCORDING TO sr_J MEASURE. (NOTE: log_{10} IS USED AS THE log FUNCTION)

Wikipedia article	sr_J
Kraków	0.5054
Uniwersytet Warszawski	0.4853
Łódź	0.4753
Poznań	0.4740
II wojna światowa	0.4707
Wrocław	0.4595
Powstanie warszawskie	0.4561
1945	0.4449
Lublin	0.4446
Lwów	0.4440
Gdańsk	0.4423
Polska	0.4409
Paryż	0.4327
2007	0.4302
2006	0.4285

TABLE IV
ARTICLE PROBABILITIES OF *zamek* LINK ACCORDING TO THE POLISH WIKIPEDIA. (THE ARTICLES WITH LESS THAN 2 LINK OCCURRENCES ARE NOT SHOWN.)

Wikipedia article	**# of occurrences**	**P**
Zamek (*castle*)	425	0.698
Zamek (*of rifle*)	60	0.099
Zamek w Bydgoszczy	28	0.046
Zamek w Bolkowie	11	0.018
Zamek w Szydłowcu	5	0.008
Zamek Świny	4	0.007
Zamek Kapituły Warmińskiej ...	4	0.007
Zamek (*for locking*)	4	0.007
Zamek Królewski w Poznaniu	3	0.005
Zamek w Malborku	2	0.003
Zamek w Rzeszowie	2	0.003
Zamek w Suchej Beskidzkiej	2	0.003
Zamek w Kowalu	2	0.003
Zamek Królewski na Wawelu	2	0.003
Zamek w Edynburgu	2	0.003

C. Sense probability

Table IV shows an example of articles that are referred to by *zamek* which is highly ambiguous according to the way people use it in Wikipedia. The Polish-English English-Polish Oxford/PWN dictionary [16] registers the following English translations for *zamek*:

- lock (combination lock, digital lock, spring lock)
- lock (of a rifle), flintlock
- castle

All of them are present in Table IV. There are also many other articles – describing various castles in Poland and United Kingdom (the last one). Given the total number of occurrences of the *zamek* link, one can compute the probability of the senses as maximum likelihood estimations.

What might be observed is the low position of the first sense of *zamek* provided in the bilingual dictionary. According to the way the dictionaries are constructed, we may assume that this meaning was identified as the most probable by its authors. As a result there seems to be a large discrepancy between the results obtained from Wikipedia and the knowledge of linguists. Still, it is not as surprising, if we consider the way Wikipedia is constructed – there are hundreds of articles describing individual castles, which will probably refer to the *castle* sense of *zamek*. On the other hand *zamek* in the *lock* sens does not have such individuals referring to it.

To sum up – although we can easily estimate the sense probability of various terms, the results might be highly skewed by the dominating contents of Wikipedia.

TABLE V
NUMBER OF OCCURRENCES AS LINK, TOTAL NUMBER OF OCCURRENCES AND *link probability* MEASURE FOR SEVERAL TERMS USED AS LINKS.

Term	**Category**	**# as link**	**# all**	**P**
Jerzy Buzek	*person*	108	236	0.458
Katowicach Giszowcu	*district*	1	4	0.250
Kraków	*city*	2756	29305	0.094
Miechów	*city*	93	359	0.259
Polska	*coutry*	3720	32164	0.116
Utraty	*river*	33	82	0.402
internetowy	*adj*	4	1281	0.003
literatury	*noun*	291	10472	0.028
małego	*adj*	2	3083	0.001
miastem	*noun*	121	8122	0.015
nie	*part/pron*	3	714190	0.000
polskiego harcerstwa	*adj+noun*	1	47	0.021
sascy	*adj*	5	46	0.109
ulica	*noun*	149	5776	0.026
zakopiański	*adj*	4	51	0.078
zamek	*noun*	609	8787	0.069

D. Link probability

The last important feature computed out of the links found in Wikipedia is the link probability, defined in [9] (which is called *keyphraseness* in that article) as:

$$P(term|W) \approx \frac{count(D_{link})}{count(D_W)} \tag{4}$$

where:

- $P(term|W)$ – link probability
- D_{link} – the number of documents where given term is used as a link
- D_W – the number of documents where given term is found

The algorithm developed by the author uses a slightly modified version of this feature – instead of counting documents, it counts the absolute number of occurrences of the term.

Table IV-D gives values of this feature for several manually selected terms. The category indicates the grammatical category of the words, except for proper names (starting with capital letters), where it indicates their semantic category. It might be observed that proper names have large link probability on average. On the other hand – popular adjectives such as *małego* (*small*) and *internetowy* (*internet*) have rather low link probability. The link probability for a particle/pronoun *nie* is negligible. The highest link probability for non-proper-names is registered for *sascy* adjective, which refers to *Saxons*.

The applicability of the link probability to identify key words, observed in the above examples was proven in [9], by comparison with other measures such as $tf \cdot idf$ and χ^2.

V. Disambiguation algorithm

The structure of the disambiguation algorithm used in Wikipedia Miner is as follows [7]:

1) the terms that have only one meaning are recognized in the document
2) the terms from the previous step are weighted according to their:
   - *average semantic relatedness* to other unambiguous terms
   - *link probability*
3) the ambiguous terms are disambiguated using a decision tree, taking into account several features of the candidate senses

The approach represented by the algorithm might be called a bag-of-senses, since it is similar to the bag-of-words approach in ignoring the order of the words and other syntactic features. The primary difference is that the senses of the terms are well defined (i.e. they are the senses of concepts described by the Wikipedia articles), while words (strings of characters in fact) are usually ambiguous. The secondary difference is that the terms might span several words whose meaning is not necessarily compositional.

The minimum requirement of the algorithm for a successful disambiguation is the presence of one unambiguous term. However in such a case the disambiguation results will be poor, unless the recognized term is central to the meaning of the senses of the ambiguous terms. So it should be noted that the quality of the results is much dependent on the presence of unambiguous terms that capture the topic of the document well. Such a requirement might be hard to meet in short passages of text. On the other hand the time complexity of the algorithm is proportional to the square of the text length (more precisely: the number of unambiguous terms), so these features compete with each other.

A. Unambiguous terms

The detection of unambiguous terms is performed using only the names of the internal links found in Wikipedia. A link is considered unambiguous if there is only one article used as the target of the link. So even if the senses that are discarded in the disambiguation procedure due to the minimal required sense probability would leave only one valid sense, the term with such a sense is regarded ambiguous.

B. Weighting

If we consider the context used to disambiguate the remaining terms, we might easily come to an idea that not all terms are equally important for the disambiguation. Some terms might be central to the topic of the document and as such their weights should be properly amplified. Other terms might be marginal, so they should have their weights damped. Milne and Witten proposed [7] to use two features for computing the weights of the terms – their average relatedness to other unambiguous terms and their link probability. The actual weight is an average of these two values. This weighting schema was used in the described algorithm without modification, with the remark, that it uses the sr_J relatedness measure.

C. Classification features

The original algorithm [7] uses the following feature for training the disambiguation classifier:

- *average weighted semantic relatedness* of the candidate sense to the senses represented by the unambiguous terms
- *probability* of the candidate sense
- *context goodness* – the quality of the disambiguation context

By using average weighted semantic relatedness computed against the unambiguous terms the sense that is most related to them should be selected. On the other hand – it might be the case that there are two or more senses that seem to be equally related. Then the one with a higher *a priori* probability should be selected.

The context goodness is registered to help the machine learning algorithm decide if the relatedness or commonness of the sense is more important. It is defined as the sum of the weights assigned to the unambiguous terms with the intention that a well defined context will favor relatedness, while weakly defined context will favor *a priori* probabilities of the senses. This assumption stems from the fact that a classifier that always selects the most probable senses performs quite well.

The observation that was made by the author concerns the two first features used by Milne and Witten. It is observed that the distribution of the *a priori* probabilities of the senses changes considerably from one term to another. There are terms that have one dominating sense and others with senses more equally distributed. In the first case, the *a priori* probability might be as high as 0.9, while in the second it might be 0.6 or even lower. On the other hand, there might be senses that are very improbable according to Wikipedia (e.g. the *lock* sense of *zamek*), but occupy relatively hight position when ranked. The advocacy for relatedness is not as direct, but it is clear that in some contexts the top senses will have very high and some times very low average relatedness.

It should be stressed that these two features do not require any additional preprocessing and are trivial to compute, so they will not impact the performance of the algorithm. As a result the author added the following features to these proposed by Milne and Witten:

- *rank of semantic relatedness* of the candidate sense among all the candidate senses
- *rank of probability* of the candidate sense among all the candidate senses
- *link probability* of the term

The link probability is another feature directly available, yet not considered by the authors. Its role is similar as the context goodness – it might help to choose between relatedness and commonness of the sense.

D. Examples selection

The method of obtaining training examples was first described in [9] – it uses the links manually defined by Wikipedia

authors as ground truth for the algorithm. It is reported that they are not always correct, but most of them lead to articles that properly describe the contextual meaning of the linked terms. So by simple extraction of these links from Wikipedia articles one can obtain hundreds of thousands of positive training examples. The negative examples are generated on the basis of the ambiguous terms – all the articles but the one linked in Wikipedia are transformed into negative examples.

It should be noted that in the version of Wikipedia used by the author (from 22^{th} July, 2011, containing 3.6M articles excluding redirects and disambiguation pages) for one positive example there are 34 negative examples on average. As a result the training set is much imbalanced.

In [7] the selection of training, tuning and testing examples was as follows – the authors randomly selected 700 articles having at least 50 internal links and divided them into three groups: 500 for training, 100 for tuning and 100 for testing. This selection scheme was much modified by the author. Defining only the minimal threshold with moderately hight value (50) means that the articles with hundreds or even thousands of links might be selected. This must be contrasted with the predicted application of the algorithm – term disambiguation in a document of moderate size (say several paragraphs) or even one paragraph of text. In such a context the probability of finding hundreds or thousands of **unambiguous** terms is very low. Ideally the algorithm should perform well even if only a few such terms were detected.

To test the performance of the algorithm in such a setting, the author imposed much more restrictive constraints on the articles – they might contain from 5 up to 100 links. The upper bound is still quite high, but much more realistic than in the original experiment. The way the examples are split into training, tuning and testing sets was the same. However the number of articles was determined differently – all the articles that met the constrains were selected from articles with ids from preselected Wikipedia database id range: 2000-12000 for English and 40000-80000 for Polish. As a result approximately 3 millions of training examples were generated for the English Wikipedia and approximately 1 million for the Polish Wikipedia.

Although the range for Polish was larger, the number of examples was smaller, since in the Polish Wikipedia (from 8^{th} of March, 2011, containing 800K articles, excluding redirects and disambiguation pages) the term ambiguity is much lower and for each positive examples „only" 12 negative examples were generated on average.

E. Classifier training

The authors of [7] evaluated several machine learning algorithms (Naive Bayes [17, p. 499], C4.5 decision tree [8] and SVM [18]) against their performance in term disambiguation. It turned out that C4.5 performed the best, so the same algorithm was employed in this version of the algorithm. Unlike Milne and Witten, the author used the original implementation of the algorithm by Quinlan – the training examples were converted to vector features and exported to text file which was then sent to the `c4.5` program.

F. Evaluation

The evaluation of the algorithm was performed on the testing data obtained from the internal link structure of Wikipedia. This is in concert with the statement that this research is an extension of the work of Milne and Witten [7]. Definitely an evaluation outside of Wikipedia context would be valuable (especially in the case of Polish), but there are no gold standard disambiguation data for Polish available and the author did not have resources to create such data. So it should be stated that the results does not reflect the real-word performance of the algorithm – they should be perceived as its upper bound.

The evaluation performed in the original experiment was similar, but covered also recreation of the links for Wikipedia articles with the markup stripped as well as human evaluation of the links provided for non-Wikipedia articles. Regarding the evaluation that overlap in both experiments – there was one peculiarity in the original experiment that was not explicitly stated in the article [7] – the reported values for precision, recall and F_1-measure were computed for ambiguous **and unambiguous** terms[2].

The evaluation in this work reports the performance of the algorithm only for ambiguous terms, since there is nothing to disambiguate, when a term is unambiguous.

TABLE VI
PERFORMANCE OF THE ORIGINAL DISAMBIGUATION ALGORITHM [7]. THE THRESHOLD FOR THE MINIMAL NUMBER OF LINKS IN THE ARTICLES WAS 50. THE RESULTS INCLUDE UNAMBIGUOUS TERMS.

	precision	recall	F_1-measure
Random sense	50.2	56.4	53.1
Most frequent sense	89.3	92.2	90.7
sr_G	**98.4**	**95.7**	**97.1**

TABLE VII
PERFORMANCE OF THE ORIGINAL ALGORITHM AND ITS IMPROVED VERSION FOR THE ENGLISH WIKIPEDIA. THE THRESHOLD FOR THE MINIMAL NUMBER OF LINKS IN THE ARTICLES IS 5 AND FOR THE MAXIMAL IS 100. THE RESULTS DO NOT INCLUDE UNAMBIGUOUS TERMS.

	precision	recall	F_1-measure
Random sense	39.1	20.8	27.2
Random sense with P > 0.5%	44.2	45.1	44.6
Most frequent sense	82.8	84.6	83.7
sr_G	83.5	84.4	84.0
sr_G + new features	83.3	85.0	84.1
sr_J	87.2	93.0	90.0
sr_J + new features	**90.5**	**94.4**	**92.4**

VI. RESULTS

The results of the original experiment [7] with two baselines are reported in Table VI. The values are very high, but it should be noted that the articles selected for evaluation contained at least 50 links and the results cover unambiguous terms.

The results of the original algorithm with the enhancements described in this article for the English Wikipedia are

[2]This was confirmed by one of the authors in private correspondence.

TABLE VIII
PERFORMANCE OF THE ORIGINAL ALGORITHM AND ITS IMPROVED VERSION FOR THE POLISH WIKIPEDIA. THE THRESHOLD FOR THE MINIMAL NUMBER OF LINKS IN THE ARTICLES IS 5 AND FOR THE MAXIMAL IS 100. THE RESULTS DO NOT INCLUDE UNAMBIGUOUS TERMS.

	precision	recall	F_1-measure
Random sense	39.7	26.4	31.7
Random sense with P > 0.5%	47.0	47.3	47.2
Most frequent sense	81.6	82.2	81.9
sr_G	82.5	83.5	83.0
sr_G + new features	84.9	83.2	84.0
sr_J	85.4	89.8	87.6
sr_J + new features	**90.4**	**93.0**	**91.7**

presented in Table VII. Table VIII contains the performance results obtained for the Polish Wikipedia. The *random sense with P > 0.5%* indicates the baseline that was obtained by selecting a random sense out of senses that have at least 0.5% *a priori* probability (cf. IV-C).

sr_G indicates the performance of the algorithm with NGD-inspired measure, while sr_J indicates its performance with Jaccard coefficient-inspired measure. The performance of the algorithm with inclusion of the new features proposed by the author (cf. V-C) is indicated by + *new features*.

Comparison of the baselines reveals that the described experiment setting is in fact different from the original experiment. By excluding unambiguous terms from the testing set the baselines dropped by several percentage points. Still it seems to be strange that if there are 35 different senses for ambiguous terms in the English Wikipedia, the first baseline is so high. This is due to the fact, that the average number of senses does not reveal the distribution of that feature. The mode of the number of senses of ambiguous terms in the English Wikipedia is 2, while the median is 8. As a result many of the disambiguation problems are reduced to the selection of one sense out of two.

The second difference that is easily spotted is the significant drop in performance of the original algorithm. The difference is so big that it raises concerns about the correctness of the experiment. However the same as with the baselines, the original performance was much affected by the inclusion of unambiguous terms in the results. But still, the difference between the most probable sense baseline and sr_G is vary small and there is no significant improvement for this semantic relatedness measure when new features are included. This might be due to some important feature of the algorithm, that was not made clear enough in [7]. But to the best knowledge of the author, there is no such feature. The most probable explanation is that the drop in performance is caused by the reduction of the number of links in the training and testing sets, making the problem much harder to tackle.

With that in mind we may observe that changing the semantic relatedness measure causes significant gain in the recall of the algorithm, besides modest gain in its precision. This effect is observed both for the English and the Polish Wikipedia. On the other hand the new features have more impact on the precision of the algorithm and this result is also consistent for the English and the Polish Wikipedia. As a conclusion it is justified to say that the improvements proposed by the author positively impact the performance of the word sense disambiguation algorithm.

The final remark concerns the comparison between the results for the English and the Polish Wikipedia. As it was stated, the gain in performance is consistent in both Wikipedias. There is also a little difference between the recall of the algorithm for the English and the Polish Wikipedia (1.4 percentage point). It is probably due to the difference in their sizes – English Wikipedia is more than 4 times larger in terms of the number of articles (3.6 millions vs. 800 thousands) and 8 times larger in terms of the dump size (32GB vs 4GB). Even these sheer size differences, the algorithms performs almost equally well.

VII. CONCLUSIONS

The motivation for this article was a construction of a word sense disambiguation algorithm for the Polish language, that will be a part of a semantic relation extraction framework. As a side effect its improved version was constructed, that performs much better than the original algorithm. Still a further assessment concerning real-word data has to be performed. It should show if Wikipedia contains enough data to perform disambiguation of inflectional language such as Polish equally well as positional such as English.

The author has also other ideas, such as asymmetric semantic measure, iterative as well as hybrid Wikipedia- and ontology-based disambiguation algorithm that may outperform the results presented here. They will be assessed in the further research.

REFERENCES

[1] D. Milne, "An open-source toolkit for mining Wikipedia," in *Proc. New Zealand Computer Science Research Student Conference*, vol. 9, 2009.
[2] C. Fellbaum, *WordNet: An Electronic Lexical Database*. MIT Press, 1998.
[3] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "DBpedia: A nucleus for a web of open data," *The Semantic Web*, pp. 722–735, 2007.
[4] F. Suchanek, G. Kasneci, and G. Weikum, "Yago: a core of semantic knowledge," in *Proceedings of the 16th international conference on World Wide Web*. ACM, 2007, pp. 697–706.
[5] I. Witten and D. Milne, "An effective, low-cost measure of semantic relatedness obtained from Wikipedia links," in *Proceeding of AAAI Workshop on Wikipedia and Artificial Intelligence: an Evolving Synergy, AAAI Press, Chicago, USA*, 2008, pp. 25–30.
[6] R. Cilibrasi and P. Vitanyi, "The google similarity distance," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 19, no. 3, pp. 370–383, 2007.
[7] D. Milne and I. Witten, "Learning to link with Wikipedia," in *Proceeding of the 17th ACM conference on Information and knowledge management*. ACM, 2008, pp. 509–518.
[8] J. Quinlan, *C4.5: programs for machine learning*. Morgan Kaufmann, 1993.
[9] R. Mihalcea and A. Csomai, "Wikify!: linking documents to encyclopedic knowledge," in *Proceedings of the sixteenth ACM conference on information and knowledge management*, 2007, pp. 233–242.
[10] M. Lesk, "Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone," in *Proceedings of the 5th annual international conference on Systems documentation*. ACM, 1986, pp. 24–26.
[11] R. Mihalcea, "Using wikipedia for automatic word sense disambiguation," in *Proceedings of NAACL HLT*, vol. 2007, no. April. Association for Computational Linguistics, 2007, pp. 196–203.

[12] P. Mendes, M. Jakob, A. García-Silva, and C. Bizer, "Dbpedia spotlight: shedding light on the web of documents," in *Proceedings of the 7th International Conference on Semantic Systems*. ACM, 2011, pp. 1–8.
[13] C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, and S. Hellmann, "DBpedia-A crystallization point for the Web of Data," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 7, no. 3, pp. 154–165, 2009.
[14] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American society for information science*, vol. 41, no. 6, pp. 391–407, 1990.
[15] P. Jaccard, "Étude comparative de la distribution florale dans une portion des Alpes et des Jura," *Bulletin del la Société Vaudoise des Sciences Naturelles*, vol. 37, pp. 547–579, 1901.
[16] "Wielki Multimedialny Słownik Angielsko-Polski Polsko-Angielski Oxford/PWN," 2004.
[17] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Prentice Hall, 2010.
[18] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.