Research Article Open Access

Eniafe Festus Ayetiran* and Kehinde Agbele

An optimized Lesk-based algorithm for word sense disambiguation

https://doi.org/10.1515/comp-2018-0015
Received May 8, 2018; accepted August 24, 2018

Abstract: Computational complexity is a characteristic of almost all Lesk-based algorithms for word sense disambiguation (WSD). In this paper, we address this issue by developing a simple and optimized variant of the algorithm using topic composition in documents based on the theory underlying topic models. The knowledge resource adopted is the English WordNet enriched with linguistic knowledge from Wikipedia and Semcor corpus. Besides the algorithm's efficiency, we also evaluate its effectiveness using two datasets; a general domain dataset and domain-specific dataset. The algorithm achieves a superior performance on the general domain dataset and superior performance for knowledge-based techniques on the domain-specific dataset.

Keywords: optimized Lesk, distributional hypothesis, topic composition

1 Introduction

The computational complexity associated with most word sense disambiguation algorithms is one of the major reasons why they are not being fully employed in most real-life applications. Agirre and Edmonds [1] identified three major approaches to word sense disambiguation; supervised, unsupervised and knowledge-based approaches. Supervised approaches rely on hand-tagged examples on which algorithms are trained and are known for their superior performance over unsupervised and knowledge-based approaches. However, they require large amount of training data which must be repeated for specific cases each time they are required. Training is also rigorous and time consuming. Unsupervised approaches, on the other hand, are self-reliant without the use of hand-tagged examples. The rigour involved in developing training set and the need for repetition for different cases make supervised approaches unappealing for several real-life applications such as text categorization, information retrieval, machine translation among others. Knowledge-based approaches primarily use dictionaries, thesauri, and lexical knowledge resources for word sense disambiguation without the need for any corpus evidence as applicable in supervised approaches. Knowledge-based techniques include graph-based methods which rely on the interconnection of semantic networks available in several lexical resources and overlap-based methods popularly called Lesk-based algorithms which originated from the original Lesk algorithm [2]. Lesk-based algorithms rely on the overlap of words between the definitions of a target word and words in context in order to determine the sense of the target word.

Algorithms based on the original Lesk algorithm are a popular and effective family of knowledge-based techniques for word sense disambiguation. Several algorithms based on the original Lesk algorithm have been developed over the years, including the adapted version which initiated the adjustability of the algorithm to fine-grained lexical resources such as the WordNet. These algorithms are generally known to be computationally costly because of the combinatorial growth of comparisons required of the several candidate senses associated with polysemous words available in different lexical resources. However, the variants of the algorithm that have been proposed over the years focused majorly on improving its effectiveness rather than improving its efficiency. A simplified variant of the algorithm attempts to solve the combinatorial explosion of Lesk-based algorithms by computing overlaps between the definition of candidate senses of the target word and the context words, it however, does not take into account the definitions of the senses of the context words. In agreement with [3], definitions are an important component in determining the meanings of words since they make distinctions more clear among polysemous words through a description of each of the senses of a word. This makes the simplified Lesk algorithm prone to poor coverage and

***Corresponding Author: Eniafe Festus Ayetiran:** Department of Mathematics & Computer Science, Elizade University, Ilara Mokin, Nigeria; E-mail: eniafe.ayetiran@elizadeuniversity.edu.ng
Kehinde Agbele: Department of Mathematics & Computer Science, Elizade University, Ilara Mokin, Nigeria;
E-mail: kehinde.agbele@elizadeuniversity.edu.ng

consequently poor recall as a result of information sparsity. The main advantage of our algorithm is that it takes sense description into account and computes similarity for each candidate sense in a single operation. That is, for an n number of senses belonging to a target word, there are exactly n number of comparisons. This kind of growth rate in algorithm analysis is linear, in contrast to the exponential growth of comparisons required of the other variants of the Lesk algorithm with the exception of the simplified Lesk.

In our algorithm, instead of combinatorial comparisons among candidate senses, we model the algorithm as a topic-document relationship based on the theory of topic models. The main idea underlying our algorithm stems from distributional hypothesis [4] on which Lesk-based algorithms generally rely. The hypothesis states that *words are similar if they appear in similar contexts*. The main theoretical footing on which our work stands is that assuming the linguistic information of all the context words made available by lexical resources are modeled as a document, and the ones provided by each of the candidate senses of the target word as topics based on the theory underlying topic models, then if the distributional hypothesis is valid, then the topic representing the correct sense of the target word should have the highest topic composition in the document. Due to the information sparsity problem predominant in overlap-based methods, we follow the work of [5] which enriches glosses of candidate senses in WordNet with additional knowledge by extending them with their corresponding wikipedia definitions obtained from BabelNet. We further enrich our algorithm with corpus knowledge from the Semcor corpus [6]. The organization of the paper is as follows: Section 2 discusses related work. We describe the optimized Lesk-based algorithm using topic composition in Section 3. Section 4 evaluates and discusses the results while Section 5 concludes the paper.

2 Related work

Our algorithm relies on the original Lesk algorithm [2] and its variants. Cowie et al. [7] presented a variation of the original Lesk algorithm called simulated annealing. In their work, they designated a function E that reflects the combination of word senses in a given text whose minimum should correspond to the correct choice of word senses. For a given combination of senses, all corresponding definitions from a lexicon are collected, and each word appearing at least once in these definitions receives a score equal to its number of occurrences. Adding all these scores together gives the redundancy of the text. The E function is then defined as the inverse of redundancy. The goal is to find a combination of senses that minimizes this function. To this end, an initial combination of senses is determined, and then several iterations are performed, where the sense of a random word in the text is replaced with a different sense, and the new selection is considered as correct only if it reduces the value of the function E. The iterations stop when there is no change in the configuration of senses. The algorithm is still complex computationally as it involves traversing a multi-path graph looking for shortest route to destination.

The Adapted Lesk algorithm [8] adjusts the original Lesk algorithm to a lexical resource, the English WordNet, by computing maximum overlap between glosses of candidate senses belonging to a target word and glosses of candidate senses of context words including the semantic relations in a combinatorial fashion based on prior tagged part-of-speech as discussed in [9]. In their work, a limited window size of the context words was used by considering only the immediate words before and after the target word. The algorithm takes as input an instance in which the target word occurs, and produces the sense for the word based on information about it and a few immediately surrounding content context words. The choice of sense is finally determined based on the maximum score achieved by computing the cumulative scores obtained from individual combinations of several candidate senses. Kilgarriff and Rosenzweig [10] in a simplified algorithm use only the context words in isolation to compute similarity among candidate senses of the target words without recourse to the definitions of senses of the target words. Ponzetto and Navigli [5] developed an extended version of the Lesk algorithm through enrichment of glosses of WordNet senses with corresponding Wikipedia definitions by using exhaustively all words in the context window of a target word. They achieved this by first mapping WordNet senses with corresponding Wikipedia terms. Their algorithm shows significant improvement in performance over the use of WordNet glosses in isolation. Basile et al. [3] in a similar fashion developed another version based on the distributional semantic model using BabelNet such that the algorithm can use all or part of the context words. Each sense in BabelNet is enriched with semantic relations using the "getRelatedMap" available in BabelNet API. In other works, Ayetiran et al. [11] & Ayetiran and Boella [12] developed a hybrid algorithm which combines a variant of the Lesk algorithm with the Jiang & Conrath similarity measure [13]. The main crux of their work is the resolution of conflict in cases where their Lesk-based algorithm and the Jiang & Conrath measure chose conflicting senses. This was done by computing similarity between the glosses of the conflicting senses and the content context words in the text.

Our algorithm is more closely related to the simplified Lesk algorithm [10] but leverages on the importance of word sense definitions using topic composition. Like other variants of the Lesk algorithm with the exception of the simplified Lesk, we use glosses of senses belonging to the content context words using the distributional hypothesis and the theory underlying topic models as a foundation. It also enriches WordNet glosses with Wikipedia definitions and corpus knowledge in the Semcor corpus. In a sense, our algorithm uses corpus information and statistics indirectly since the SemCor corpus contains these two properties.

3 Optimizing an adaptation of the Lesk algorithm

We mentioned in Section 1 that the combinatorial explosion in variants of the Lesk algorithm is a result of the polysemous nature of some words, each represented by candidate senses within the lexical resources used for word sense disambiguation. The methodology we devised consists of three phases. First, we build documents from linguistic information derived for the context words in the lexical resources and the corpus. Secondly, we build topics from linguistic information derived from the candidate senses of the target words. Finally, we obtain the composition of each topic in a document, each represented by the candidate senses of the target word. Section 3.3 to section 3.5 provide a detailed description of these phases.

3.1 Knowledge resources

- **WordNet.** WordNet [14] is a manually-constructed lexical knowledge system. The basic object in WordNet is a set of synonyms called a synset. By definition, each synset in which a word appears is a different sense of that word. There are four main divisions in WordNet, one each for nouns, verbs, adjectives and adverbs. Within a division, synsets are organized by the lexical relations defined on them. For nouns, the lexical relations include hypernymy/hyponymy (IS-A relation) and three different meronymy/holonymy (PART-OF) relations. The verb also includes hypernymy/hyponymy, troponymy and other relations like entailment, causes etc. The IS-A relation is the dominant relation, and organizes noun and verb synsets into a set of hierarchies while the adjectives and adverbs are organized as clusters. WordNet is the most widely adopted standard English lexicon.
- **BabelNet.** BabelNet [15] serves as an "encyclopedic dictionary" by merging WordNet, Wikipedia and other multilingual lexical resources. Wikipedia is a multilingual web-based encyclopedia. It is a collaborative open source medium edited by volunteers to provide a very large wide-coverage repository of encyclopedic knowledge. Each article in Wikipedia is represented as a page (referred to as Wikipage) and presents information about a specific concept or named entity. BabelNet provides a mapping of WordNet senses with their corresponding wikipedia definitions. Through this mapping, an extension of WordNet glosses is made possible through which we have enriched the linguistic information in both documents and topics.
- **SemCor.** SemCor [6] is a subset of the Brown corpus that contains about 362 texts comprising over 200.000 words which have been manually tagged with part-of-speech and senses from WordNet. It is a good resource for projects requiring sense annotations of words and has been used mostly in corpus-based similarity methods for word sense disambiguation.

3.2 Topic models and composition

Topic models [16–20] provide a powerful tool for analyzing large text collections by representing them as a low dimensional set of topics. They are based upon the idea that documents are mixtures of topics, where a topic is a probability distribution over words. A topic model is a generative model for documents; it specifies a simple probabilistic procedure by which documents can be derived. Each topic is a multinomial distribution over words and the highest probability words summarize the subjects in the document collection.The major strengths of topic models are dimensionality reduction and thematic semantic information extraction. Topic models have been applied in different areas including text categorization, word sense disambiguation, information retrieval, sentiment analysis, data mining, document summarization etc. The combination of probabilities in a topic gives the topic weight. Interpretability of individual topics provided by their probability distribution over words which depicts a cluster of correlated words is a major distinct advantage of representing the documents with probabilistic topics over a purely spatial representation. In other words, topic models reveal the subject of discussion in documents, for instance in communications like e-mail, tweets etc.

A generative model for documents is based on simple probabilistic sampling rules that describe how words in documents might be generated on the basis of latent variables. When fitting a generative model for a collection, the goal is to find the best set of random variables that can explain the observed words in the documents. The generative process does not make any assumptions about the order of words as they appear in documents. The only information relevant to the model is the number of times words appear in the documents. This is known as the bag-of-words model, and is common in many statistical models of language including latent semantic indexing (LSI) and probabilistic latent semantic indexing (pLSI). However, word order information in some cases might contain important clues to the content of a document. Blei et al. [16] introduced an extension of the topic model that is sensitive to word order and automatically learns the syntactic as well as semantic factors that guide word choice.

Probabilistic models use the same idea as generative models; that a document is a mixture of topics, with a slightly different statistical assumptions. Let us take $P(z)$ to be the probability distribution over topics z in a particular document and $P(w|z)$ to be the probability distribution over words w in a topic given a set of topics z. Each word w_i in a document is generated by first sampling a topic from the topic distribution and then choosing a word from the topic-word distribution, where i refers to the ith word token. We denote $P(z_i = j)$ as the probability that the jth topic was sampled for the ith word token and $P(w_i|z_i = j)$ as the probability of word w_i under topic j. The model specifies distribution over words within a document as presented in equation 1:

$$P(w_1) = \sum_{j=1}^{T} P(w_i|z_i = j)P(z_i = j) \tag{1}$$

where T is the number of topics, $P(w_i|z_i = j)$ refer to the multinomial distribution over words at index $i(i > 0)$ for topic j and $P(z_i = j)$ is the multinomial distribution over topics for document d. Let us take D to be a document collection consisting of several documents with each document d consisting of Nw word tokens where N is the total number of word tokens, that is, $N = \sum Nw$. The parameters $P(w_i|z_i = j)$ and $P(z_i = j)$ respectively, indicate which words are important for which topic and which topics are important for a particular document.

3.3 Document development

A document is developed at each sentence level in which a target word occurs. Let us consider a text T containing sentences of size m, $T = s_1, s_2, ..., s_m$ and each sentence, $s_i \in T$ containing words W of size n, $W = w_1, w_2 ....., w_n$. A word $w_i \in W$ is designated to be disambiguated (target word) and all other words $w_j \in W, j \neq i$ are designated as the context words. In the first stage of the document development, the glosses of all the synsets of all the content context words, together with synsets that are semantically related in WordNet which includes synonyms, hypernyms, hyponyms, meronyms, holonyms, antonyms etc.,[1] are harvested to form the first initial document for that context using a context window [2]. In the second stage, we lookup BabelNet for the corresponding wikipedia definitions of the WordNet senses (including that of their semantic relations) which we have used in the first stage to enrich the document with more linguistic information and from which the second initial document is derived. In the third stage, the third initial document is developed by harvesting all the sentences where the main synsets of each of the content context words appear in the Semcor corpus. A raw document is generated through the accumulation of all the initial documents built at each of the development stages as presented in equation 2.

$$d = \sum_{i=1}^{n} d_i \tag{2}$$

where d is the derived raw document from the accumulation of the component documents and d_i are the component documents and $n = 3$, the total number of knowledge sources. The raw document is a bag of words resulting from the accumulation of the linguistic information from the different sources. The final document is a set of words derived from the bag of words. Therefore, for a target word, there is a single final document $d = \{w_1, w_2, ..., w_j\}$ obtained from the initial bag of words representation of the document.

1 Note that Banerjee and Pedersen (2003) recommend that all the relations may not be helpful and that optimum choice of relations for usage depends on the specific application. However, in this particular application, we need the glosses of all the relations to a word sense in order to fully characterize the word sense since the accumulation of the glosses of these relations is what is required to build the topics and documents since full characterization is required to compute the topic composition in a document that has been built using the theory of topic models. In fact, using specific relations results in inferior performance.

2 The context window in this case is the whole sentence.

3.4 Topic development

Topic development follows a similar fashion as obtainable in the document development phase. However, it is built at the word sense level of each target word. Equation 3 shows the accumulation of the initial topics from the 3 knowledge sources.

$$t = \sum_{i=1}^{n} t_i \tag{3}$$

where t is the derived final topic and t_i are the component topics and $n = 3$, the total number of knowledge sources. The raw topics are also a bag of words derived from the 3 knowledge sources from which a set of words is derived to form intermediate topics. The final topics are a set of words in the intermediate topics minus words which appear in more than one of the intermediate topics [3]. Excluding the multiply-occurring words in the intermediate topics helps in avoiding poor characterization for each sense of a target word. For each target word, there are n number of topics corresponding to the number of senses the target word has in the WordNet. Therefore, for a target word, there are n topics t, $t = \{w_1, w_2, ..., w_j\} - f_{j=1}^{n}$ obtained from the initial bag of words representation of the topics, where $f_{j=1}^{n}$ are multiply-occurring words in the topics i.e. words that appear in more than one of the intermediate topics representing the senses of a target word.

3.5 Computing topic composition in the documents

The proportion of topic composition in the document can be computed using any document similarity method but we found cosine similarity most appropriate since it normalizes the different lengths of the topics. In this case, after normal document processing on both topics and the documents, such as stopwords removal and normalization which includes stemming, the topic composition (similarity) is computed using the cosine similarity between each of the final topics and documents using equation 4.

$$cos\theta = \frac{\vec{t}.\vec{d}}{||\vec{t}||\,||\vec{d}||} \tag{4}$$

where cosθ is the cosine similarity between t and d, $t.d$ is the dot product and $||t||$ and $||d||$ are the vector lengths of t and d, respectively.

3 Note that each intermediate topic represents each of the word senses belonging to a target word.

4 Evaluation

In order to see how effective our optimized Lesk algorithm is, we carry out experiments on two datasets; a general domain dataset and a domain-specific dataset. For the general domain experiment, we use the English dataset of SemEval 2013 multilingual word sense disambiguation. The domain-specific experiment uses the English dataset of SemEval 2010 all-words word sense disambiguation on specific domain. We then compare the results of our algorithm, each with the state-of-the-art systems that participated in both tasks.

4.1 Evaluation on general domain word sense disambiguation

The SemEval 2013 multilingual word sense disambiguation [21] presented tasks in English, French, German, Italian and Spanish. It used fine-grained sense distinctions for scoring systems.The dataset is tagged with both WordNet and BabelNet sense inventories. We experimented with the English dataset since our lexical resources and corpus are in English. In Table 1, we present the results of the optimized Lesk and a reproduction of the simplified Lesk [4] on the dataset. Table 2 juxtaposes the result of the optimized Lesk with that of participating systems.

Table 1: Performance of optimized and simplified Lesk Algorithms on the English Dataset of SemEval 2013 Multilingual WSD.

Algorithm	Precision	Recall	F1
optimized Lesk	**0.663**	**0.657**	**0.660**
simplified Lesk	0.644	0.377	0.476

Table 2: Performance of optimized Lesk and Participating Systems on the English Dataset of SemEval 2013 Multilingual WSD.

Team	System	Precision	Recall	F1
-	optimized Lesk	**0.663**	**0.657**	**0.660**
GETALP	WN-1	0.406	0.406	0.406
UMCC-DLSI	RUN-1	0.639	0.635	0.637
UMCC-DLSI	RUN-2	0.649	0.645	0.647
UMCC-DLSI	RUN-3	0.642	0.639	0.640
MFS		0.63	0.63	0.63

4 By "simplified Lesk" we refer to the algorithm in (Kilgariff and Rosenzweig, 2010) which is an improved variant of the original Lesk algorithm (Lesk, 1986) that has been chosen for comparison because it is the only one having the same time complexity with the hereby proposed new variant of the Lesk algorithm.

Table 3: Performance of optimized and simplified Lesk on the English Dataset of SemEval 2010 Domain-Specific WSD.

Algorithm	P	R	R Nouns	R Verbs
optimized Lesk	**0.516**	**0.504**	**0.513**	**0.478**
simplified Lesk	0.370	0.196	0.230	0.101

4.2 Evaluation on domain-specific word sense disambiguation

The SemEval 2010 all-words word sense disambiguation on a specific domain [22] was proposed as a result of new challenges posed by domain portability and NLP components. Tasks were organized in English, Chinese, Dutch and Italian. We deal with the English dataset since our knowledge resources are in English. Prior to this task, all available tasks on a specific domain use lexical-sample dataset. The exercise also scores systems using fine-grained sense distinctions. Agirre et al. [22] identify the issues specific domains pose to WSD systems: the context in which the senses occur might change; different domains involve different sense distributions and prevailing senses, some words tend to occur in fewer senses in particular domains, the context of the senses might change, and new senses and terms might be involved. They further revealed that both supervised and knowledge-based systems are affected by these issues: while the former suffer from different context and sense priors, the latter suffers from lack of coverage of domain-related words and information. Therefore, domain-specific word sense disambiguation presents an entirely new scenario in evaluation of word sense disambiguation systems. The SemEval 2010 all-words domain specific WSD had both supervised and knowledge-based systems' participation. First, we present the results of our system and a reproduction of the simplified Lesk in Table 3. Table 4 shows a juxtaposition of the performance of our algorithm and participating systems.

4.3 Discussion of results

First, we discuss the results obtained from our algorithm in relation to a reproduction of simplified Lesk which is a variant of Lesk-based family of algorithms. This is in view of the fact that the two algorithms have the same complexity in terms of running time. Results from the evaluations on the two datasets show a consistent superior performance of our algorithm over the simplified Lesk algorithm. The simplified Lesk is the only variant of Lesk-based algorithms which attempts to resolve the computational complexity as major obstacle affecting the applications of this family of algorithms in real-life applications. However, it considers only the information provided by the context of the word being disambiguated. Results from our experiments confirm the existing knowledge of its effect on performance as a result of the information sparsity problem [10]. Real-life applications of word sense disambiguation need to exploit the benefits provided by knowledge-based techniques due to the comparable minimal effort required for their development, generality of usage and reusability. However, while considering the efficiency which these techniques offer, there is also the need for adequate balance with effectiveness to cater for the overall goal of these applications. Comparisons among our system and other systems for the SemEval 2013 multilingual WSD task show that our system outperforms other systems in all the 3 evaluation metrics with a score of 0.663, 0.657 and 0.660 for precision, recall and F1 respectively. Furthermore, on the SemEval 2010 domain-specific task, it achieves superior performance among the knowledge-based techniques and the best performance on verbs along with the IIITH2-d.r.l.ppr.05 system with a recall of 0.478. The main evaluation metric for the task is recall. For the participating systems, recall measure is accompanied by a 95% confidence interval using bootstrap resampling to check the statistical significance between ranked systems if there is no overlap in confidence intervals. In all, about 29 systems participated in the domain-specific task in which a supervised system, CFILT-2 [23] achieves the overall best performance with a precision and recall of 0.570 and 0.555 ±0.024 respectively.

5 Conclusion

Real-life applications which require unequivocalness in texts have not fully taken the advantages of knowledge-based techniques for word sense disambiguation. These advantages include generality, comparable minimal development effort and reusability when compared with supervised techniques. One of the main reason for this is the computational complexity involved in implementing knowledge-based algorithms. In this work, we investigate the optimization of the computational complexity associated with a popuplar and effective knowledge-based algorithm - the Lesk-based algorithm. Our investigation reveals that the complexity can be greatly reduced while at the same time achieving high performance. Furthermore, we show that using this algorithm, linguistic knowledge can be enriched with corpus knowledge from annotated cor-

Table 4: Performance of optimized Lesk and Participating Systems on the English Dataset of SemEval 2010 Domain-specific WSD.

Rank	Participant	System ID	Type	P	R	R Nouns	R Verbs
1	Anup Kulkarni	CFILT-2	WS	**0.570**	**0.555 ±0.024**	**0.594±0.028**	0.445±0.047
2	Anup Kulkarni	CFILT-1	WS	0.554	0.540 ±0.021	0.580 ±0.025	0.426 ±0.043
3	Siva Reddy	IIITH1-d.l.ppr.05	WS	0.534	0.528 ±0.027	0.553 ±0.023	0.456 ±0.041
4	Abhilash Inumella	IIITH2-d.r.l.ppr.05	WS	0.522	0.516 ±0.023	0.529 ±0.027	**0.478 ±0.041**
5	Ruben Izquierdo	BLC20SemcorBackground	S	0.513	0.513 ±0.022	0.534 ±0.026	0.454 ±0.044
-	-	Most Frequent Sense	-	0.505	0.505 ±0.023	0.519 ±0.026	0.464 ±0.043
6	Ruben Izquierdo	BLC20Semcor	S	0.505	0.505 ±0.025	0.527 ±0.031	0.443 ±0.045
7	-	optimized Lesk	KB	0.516	0.504	0.513	**0.478**
8	Anup Kulkarni	CFILT-3	KB	0.512	0.495 ±0.023	0.516 ±0.027	0.434 ±0.048
9	Andrew Tran	Treematch	KB	0.506	0.493 ±0.021	0.516 ±0.028	0.426 ±0.046
10	Andrew Tran	Treematch-2	KB	0.504	0.491 ±0.021	0.515 ±0.030	0.425 ±0.044
11	Aitor Soroa	kyoto-2	KB	0.481	0.481 ±0.022	0.487 ±0.025	0.462 ±0.039
12	Andrew Tran	Treematch-3	KB	0.492	0.479 ±0.022	0.494 ±0.028	0.434 ±0.039
13	Radu Ion	RACAI-MFS	KB	0.461	0.460 ±0.022	0.458 ±0.025	0.464 ±0.046
14	Hansen A. Schwartz	UCF-WS	KB	0.447	0.441 ±0.022	0.440 ±0.025	0.445 ±0.043
15	Yuhang Guo	HIT-CIR-DMFS-1.ans	KB	0.436	0.435 ±0.023	0.428 ±0.027	0.454 ±0.043
16	Hansen A. Schwartz	UCF-WS-domain	KB	0.440	0.434 ±0.024	0.434 ±0.029	0.434 ±0.044
17	Abhilash Inumella	IIITH2-d.r.l.baseline.05	KB	0.496	0.433 ±0.024	0.452 ±0.023	0.390 ±0.044
18	Siva Reddy	IIITH1-d.l.baseline.05	KB	0.498	0.432 ±0.021	0.463 ±0.026	0.344 ±0.038
19	Radu Ion	RACAI-2MFS	KB	0.433	0.431 ±0.022	0.434 ±0.027	0.399 ±0.049
20	Siva Reddy	IIITH1-d.l.ppv.05	KB	0.426	0.425 ±0.026	0.434 ±0.028	0.399 ±0.043
21	Abhilash Inumella	IIITH2-d.r.l.ppv.05	KB	0.424	0.422 ±0.023	0.456 ±0.025	0.325 ±0.044
22	Hansen A. Schwartz	UCF-WS-domain.noPropers	KB	0.437	0.392 ±0.025	0.377 ±0.025	0.434 ±0.043
23	Aitor Soroa	kyoto-1	KB	0.384	0.384 ±0.022	0.382 ±0.024	0.391 ±0.047
24	Ruben Izquierdo	BLC20Background	S	0.380	0.380 ±0.022	0.385 ±0.026	0.366 ±0.037
25	Davide Buscaldi	NLEL-WSD-PDB	WS	0.381	0.356 ±0.022	0.357 ±0.027	0.352 ±0.049
26	Radu Ion	RACAI-Lexical-Chains	KB	0.351	0.350 ±0.015	0.344 ±0.017	0.368 ±0.030
27	Davide Buscaldi	NLEL-WSD	WS	0.370	0.345 ±0.022	0.352 ±0.027	0.328 ±0.037
28	Yoan Gutierrez	Relevant Semantic Trees	KB	0.328	0.322 ±0.022	0.335 ±0.026	0.284 ±0.044
29	Yoan Gutierrez	Relevant Semantic Trees-2	KB	0.321	0.315 ±0.022	0.327 ±0.024	0.281 ±0.040
30	Yoan Gutierrez	Relevant Cliques	KB	0.312	0.303 ±0.021	0.304 ±0.024	0.301 ±0.041
-	-	Random baseline	-	0.232	0.232	0.253	0.172

pora for overall effectiveness of word sense disambiguation without hurting efficiency.

Acknowledgement: Our gratitude goes to Elizade University for providing the fund through which this work was made possible. We express our deep appreciation to the anonymous reviewers for their efforts in improving the initial manuscript.

References

[1] Agirre E., Edmonds P., Word Sense Disambiguation: algorithms and applications, Text, Speech and Language Technology, 2006, 33

[2] Lesk M., Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone, In: Proceedings of the 5th ACM-SIGDOC conference, ACM, 1986, 24–26

[3] Basile P., Caputo A., Semeraro G., An enhanced Lesk word sense disambiguation algorithm through a distributional semantic model, In: Proceedings of the 25th International Conference on Computational Linguistics: Technical Papers, Dublin, Ireland, August 23-29, 2014, 1591–1600

[4] Harris Z., Distributional structure, Word, 1954, 10(23), 146–162

[5] Ponzetto S. P., Navigli R., Knowledge-rich word sense disambiguation rivaling supervised systems, In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, Uppsala, Sweden, July 11-16, 2010, 522–1531

[6] Miller G. A., Leacock C., Tengi R., Bunker R. T., A semantic concordance, In: Proceedings of the ARPA Workshop on Human Language Technology, 1993, 303–308

[7] Cowie J., Guthrie J., Guthrie L., Lexical disambiguation using simulated annealing, In: Proceedings of the 14th Conference on Computational Linguistics (COLING), ACL, 1992, 359–365

[8] Banerjee S., Pedersen T., An adapted Lesk algorithm for word sense disambiguation using wordNet, In: Proceedings of the 3rd International Conference on Computational Linguistics and Intelligent Text Processing, 2002, 136–145

[9] Banerjee S., Pederson T., Extended gloss overlaps as a measure of semantic relatedness, In: Proceedings of the 18th International Joint Conference on Artificial Intelligence, Acapulco, Mexico, August 9-15, 2003, 805–810

[10] Kilgarriff A., Rosenzweig J., Framework and results for English SENSEVAL, Computers and the Humanities, 2000, 34(1-2)

[11] Ayetiran E. F., Boella G., Caro L. D., Robaldo L., Enhancing word sense disambiguation using a hybrid knowledge-based technique, In: Proceedings of 11th International Workshop on Natural Language Processing and Cognitive Science, 2014, 15–26

[12] Ayetiran E. F., Boella G., EBL-hope: Multilingual word sense disambiguation using a hybrid knowledge-based technique, In: Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), Denver, Colorado, June 4-5, 2015, 340–344

[13] Jiang J., Conrath D., Semantic similarity based on corpus statistics and lexical taxonomy, In: Proceedings of the 10th International Conference on Research in Computational Linguistics, 1999, 19–33

[14] Fellbaum C., An electronic lexical database, MIT Press, 1998, ed.

[15] Navigli R., Ponzetto S. P., BabelNet: The automatic construction, evaluation and application of a widecoverage multilingual semantic network, Artificial Intelligence, 2012, 193(2012), 217–250

[16] Blei D. M., Ng A. Y., Jordan M. I., Latent Dirichlet allocation, Journal of Machine Learning Research, 2003, 3, 993–1022

[17] Griffiths T. L., Steyvers M., A probabilistic approach to semantic representation, In: Proceedings of the 24th Annual Conference of Cognitive Science Society, 2002

[18] Griffiths T. L., Steyvers M., Prediction and semantic association, Advances in Neural Information Processing Systems, 2003, 15, 11–18

[19] Griffiths T. L., Steyvers M., Finding scientific topics, In: Proceedings of the National Academy of Sciences, 2004, 5228–5235

[20] Hoffman T., Probabilistic latent semantic indexing, In: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1999, 50–57

[21] Navigli R., Jurgens D., Vannella D., SemEval-2013 Task 12: Multilingual word sense disambiguation, In: Proceedings of the Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: 7th International Workshop on Semantic Evaluation (SemEval 2013), Atlanta, Georgia, June 14-15, 2013, 222–231

[22] Agirre E., de Lacalle O. L., Fellbaum C., Hsieh S. K., Tesconi M., Monachini M., Vossen P., Segers R., SemEval-2010 Task 17: All-words word sense disambiguation on a specific domain, In: Proceedings of the 5th International Workshop on Semantic Evaluation, Uppsala, Sweden, July 15-16, 2010, 75–80

[23] Kulkarni A., Khapra M. M., Sohoney S., Bhattacharyya P., CFILT: Resource conscious approaches for all-words domain-specific WSD, In: Proceedings of the 5th International Workshop on Semantic Evaluation, Uppsala, Sweden, July 15-16, 2010, 421–426