Arabic Word Sense Disambiguation for Information Retrieval

MOHAMMED ALAEDDINE ABDERRAHIM and MOHAMMED EL-AMINE ABDERRAHIM, Laboratoire de Traitement Automatique de la Langue Arabe (LTALA), Tlemcen University

In the context of using semantic resources for information retrieval, the relationship and distance between concepts are considered important for word sense disambiguation. In this article, we experiment with Conceptual Density and Random Walk with graph methods to enhance the performance of the Arabic Information Retrieval System. To do this, a medium-sized corpus was used. The results proved that Random Walk can enhance the performance of the information retrieval system by achieving a mean improvement of 13%, 16%, and 12% in terms of recall, precision, and F-score, respectively.

CCS Concepts: • **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability;

Additional Key Words and Phrases: IRS, conceptual density, PageRank, Arabic WordNet, WSD, information retrieval

ACM Reference format:
Mohammed Alaeddine Abderrahim and Mohammed El-Amine Abderrahim. 2022. Arabic Word Sense Disambiguation for Information Retrieval. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* 21, 4, Article 69 (January 2022), 19 pages.
https://doi.org/10.1145/3510451

1 INTRODUCTION

Ambiguity of a word is to have more than one interpretation; it is a key feature of natural languages. **Word Sense Disambiguation (WSD)** is the problem of identifying the sense of a word within a specific context. That is, words can have different meanings (polysemy), depending on the context in which they occur. It is a key problem of **Natural Language Processing (NLP)**.

The task of WSD has received a lot of attention in English. However, the amount of work in other languages, such as Arabic, is still limited [28], despite the fact that there are half a billion native Arabic speakers.

The global importance of Arabic should provide motivation to spur additional in-depth studies. This is especially true, because the availability of tools and systems in other languages would dramatically affect the performance of downstream systems in English.

Authors' address: M. A. Abderrahim and M. E.-A. Abderrahim, Laboratoire de Traitement Automatique de la Langue Arabe (LTALA), Tlemcen University, Box 13000, Tlemcen, Algeria; emails: Abderrahim.alaa@yahoo.fr, med.amine.abderrahim@gmail.com.

2375-4699/2022/01-ART69 $15.00
https://doi.org/10.1145/3510451

There are some methods based on the metric of counting the overlaps between definitions of the words for calculating the relatedness among senses [25], where others methods are based on distance between concepts following the structure of the **Lexical Knowledge Base (LKB)** [11]. We also find method based on the distances calculated using hierarchical relations on the LKB [28, 42], as well as the methods based on semantic graphs [4].

These methods use well-known graph-based techniques to find and exploit the structural properties of the graph underlying a particular LKB. Graph-based techniques consider all the sense combinations of the words occurring on a particular context at once, and thus offer a way to analyze the relations among them with respect to the whole graph. They are particularly suited for disambiguating words in the sequence, and they manage to exploit the interrelations among the senses in the given context [4]. Graph-based WSD is performed by a graph composed of senses (nodes) and relations between pairs of senses (edges). The relations may be of several types (lexicosemantic, cooccurrence, etc.) and may have some weight attached to them. Some methods are based on WordNet as a LKB. Disambiguation is typically performed by applying a ranking algorithm over the graph, and then assigning the concepts with highest rank to the corresponding words. Given the computational cost of using large graphs like WordNet, most researchers use smaller subgraphs built on line for each target context. The main idea of the subgraph method is to extract the subgraph whose nodes and relations are particularly relevant for the set of senses from a given input context. The subgraph is then analyzed and the most relevant nodes are chosen as the correct senses of the words [4].

In Arabic language, ambiguity is present at many levels [22], such as homograph, internal word structure, syntactic, semantic, constituent boundary, and anaphoric ambiguity. Also, even to fluent native speakers, the absence of diacritics in most Arabic texts, which are used to guide pronunciation, leads to much ambiguity in the perceived meaning. Our motivation for this work comes from the robustness and the effectiveness of these methods of WSD for English.

Arabic language belongs to the Afro-Asian language group. Its writing is right to left. It is one of the six official languages of the United Nations, it is the mother tongue of more than 300 million people. Diacritization or vocalization in Arabic consists of adding a symbol (a diacritic) above or below letters to indicate the proper pronunciation and meaning of a word. The absence of the diacritization in most of Arabic electronic and printed media poses a real challenge for Arabic language understanding [1].

The Arabic language is very rich semantically and lexically. Dealing with ambiguity in Arabic is considered as the most challenging task in Arabic NLP. The ambiguity of Arabic language is due to their linguistic variation. There are two main levels of Arabic ambiguity. First, the ambiguity of words that have the same spelling, but different meanings. Their main cause is due to the fact that the majority of digital documents do not include diacritics. Second, the polysemy, which means that one word has more than one meaning [22].

In this article, we are interested to use the distance between concepts for disambiguation of Arabic texts for a best IR. The methods of **Conceptual Density (CD)** and the PageRank are implemented and tested for WSD. These methods are based on the graph extracted from **Arabic WordNet (AWN)**. The remainder of the article is organized as follows. The next section presents a brief related work about WSD. Section 3 describes the CD and CD with frequency methods for performing WSD. Section 4 gives the CD algorithm to achieve WSD. Section 5 provides the PageRank algorithm to achieve WSD. Section 6 describes the experimental setting including data preparation, evaluation and discussion. Finally, Section 7 presents the conclusion and suggestions for future work.

2 RELATED WORK

There are many approaches of WSD for the English language, and most of these approaches give good results. These approaches can be distinguished into two major kinds:

- Corpus-based approaches. These approaches are based on machine learning techniques. They acquire the necessary information to define words' senses from a corpus. They are divided into three categories: supervised approaches, semi-supervised approaches, and unsupervised approaches. The unsupervised approaches do not include any manually sense-tagged corpus to generate a sense choice for a word given its context. The learning in this approach is based on the idea that occurrences of a word that have the same sense often have similar co-occurrences. These neighboring words are grouped into clusters. These clusters are considered appropriate senses for ambiguous words [40]. The supervised and semi-supervised approaches have taken advantage of some machine learning methods to train a classifier from labeled training sets. These sets enclose some examples encoded in terms of a number of features together with their suitable sense label (or class) [50]. **Embeddings from Language Models (ELMo)** is a technique based on machine learning for word representation [35]; it is based on contextualised and token-based as opposed to type-based word representations, like word2vec or GloVe, where each word type is assigned a single vector. Compared to traditional type-based, the use of ELMO for word representation is better. ELMo is tested in Reference [19], it uses a corpus of 20 million words per language to train monolingual language models for many languages. Corpus-based approaches are outside the scope of this work.
- Knowledge-based approaches. These approaches are based on the use of external lexical resources, like dictionaries, thesauruses, or ontologies. These resources contain all the words of a language with their senses [40]. In the context of this article, we are interested in these approaches.

In the following sections, we cite some works based on Knowledge-based approaches for **Information Retrieval (IR)**.

The authors of Reference [20] disambiguate words according to their domain by using **Longman Dictionary of Contemporary English (LDOCE)**. The work cited in Reference [46] proposes to manually extend the context and the meanings of an ambiguous word by adding the words that consistently occur with the word and their meanings in context. This method is tested on LDOCE and achieved a 45% performance rate. Yarowsky's work [47], which is based on the semantic categories[1] of Roget's[2] thesaurus, disambiguates the meaning of the Grolier Multimedia Encyclopedia. This disambiguation consists of determining the semantic category from the thesaurus by associating the keywords of the target category. Reference [42] proposes a method of name disambiguation based on CD by using WordNet's synonymy and antonym relations. This method is tested on the Time[3] collection, and it had a 56% precision rate. In Reference [44] the most appropriate Synset of an ambiguous word is selected from WordNet by computing the numbers of common words between the Synset and the context. The results showed that the effectiveness of the vectors produced by this disambiguation technique was worse than word stem vectors for all five collections. Reference [3] uses a CD to enrich dictionary senses of the **Intelligent Dictionary Help System (IDHS)** with semantic tags extracted from WordNet, the authors do not report any

[1]The Roget thesaurus contains 1,024 categories of domains (ANIMAL/INSECT, TOOLS/MACHINERY, etc.), which cover the different meanings of the words.
[2]http://www.roget.org/.
[3]The Time collection consists of articles from the magazine *Time*.

experimental results. References [5, 6, 37] presents an automatic decision procedure for lexical ambiguity resolution based on the CD for names. This approach is based on WordNet. The results of the experiments have been automatically evaluated on SemCor;[4] this method achieved 47.3% of precision rate. Reference [30] disambiguates all the nouns, verbs, adverbs, and adjectives in a given text by using the senses provided by WordNet; the authors do not report any experimental results. The authors of Reference [27] use WordNet domains in WSD. They use domain labels to establish semantic relations among word senses, which can be profitably used during the disambiguation process. Reference [39] explores a fully automatic knowledge-based method that performs the noun sense disambiguation relying only on WordNet. This method is based on the CD, it achieved 81.48% of precision rate. Reference [17] uses WordNet domains and the Cambridge Advanced Learner's Dictionary to disambiguate the words in their fields. The authors used the method of CD and **CD with frequency (CDF)**. This method is tested on the SemCor corpus, they obtained a 80% of precision rate. Reference [23] uses the domains and WordNet [26] to disambiguate the words by determining the domain of the text where the ambiguous word appears. The domains and WordNet are used to the word categorization system that facilitate the disambiguation. For example, the term bank has 10 meanings, but has only 5 possible domains, because economy gathers 4 senses, geography gathers 2 senses, architecture groups 2 senses, transport groups 1 meaning, and it remains a general direction; this method achieved 79% of precision rate. Reference [32] proposes an approach based on the Macquarie[5] thesaurus for WSD. The basic idea of this approach is that the majority of occurrences of a word in a corpora have the same sense. This approach is done on a small sample of the **British National Corpus (BNC)** World Edition. The results displayed a 50% precision rate. Reference [16] disambiguates place names found in SemCor (GeoSemCor: Geometric SemCor) through the use of WordNet. In this work, the authors used the method of CDF, and compare the results with a variant of Lesk. The obtained results show that a better precision can be achieved by using a small context, whereas a greater coverage can be obtained by using large contexts. The authors of Reference [9] tried to expand queries using WSD, and then they compared the performance of a search engine before and after expanding the query. Their model is based on the user feedback for building queries. They expand a query terms by the addition of more specific synonyms and test the searching results. The results of their experiment demonstrates that the expansion of a query terms will narrow the search and it increases precision and recall. While the expansion of a query using more general synonyms decrease the precision.

The authors of Reference [24] suggested two techniques for WSD dedicated to Arabic text categorization. In the first approach, authors used the Arabic WordNet and WordNet based on term to term **Machine Translation System (MTS)**. The goal of the second technique is to identify the nearest concept for the ambiguous Arabic words based on more relationships with different concepts in the same local context. Authors have used in their experiments the **Essex Arabic Summaries Corpus (EASC)**. Several combinations of tests have been performed using the feature selection methods (CHIR and Chi-Square) and two machine learning techniques (**Support Vector Machine (SVM)** and the **Naïve Bayesian (NB)**). However, the best accuracy rate (73.2%) was achieved due to Wu and Palmer's measure, the Chi-Square to features selection and the SVM classifier.

Reference [13] proposed a method based on the WordNet (is-a) relationship to disambiguate nouns and verbs. The approach has been tested on the Muchmore[6] collection. The results presented a precision rate of more than 50%.

[4] http://web.eecs.umich.edu/~mihalcea/downloads.html#semcor.
[5] http://www.macquariedictionary.com.au/anonymous@9c9B329512906/-/p/dict/index.html.
[6] http://muchmore.dfki.de/.

References [12, 18] calculate the distributional similarity between definitions and the context of the word to disambiguate, this method achieved a precision rate equal to 64.2%.

There are other works that use the conceptual graphs extracted from the semantic resources to find the best sense; these works are based on the random walk approaches on the graph [4, 7, 8, 33, 34, 36, 41, 43, 45].

At first, these systems typically create a graphical (mathematical) representation of the input text and then exploit different graph-based algorithms for WSD on the given representation, like the PageRank algorithm [7].

In this type of disambiguation, the graph is constructed from the texts where the nodes are the senses of the words of texts, and the relations between these nodes are extracted from a semantic resource. The links in the graph are weighted with the execution of the algorithm.

A variation of the PageRank method proposed in Reference [8] is called **Personalized PageRank (PPR)**. The basic idea behind the adoption of PPR is to create a personalized vector that expresses the contexts of all the words targeted by disambiguation. This method improves the complexity of the methods presented previously [7], because it makes it possible to contextualize the behaviours of PageRank on a sentence, without asking for a different graph; in this way the WordNet graph is always adopted to a word or phrase.

In addition, it is possible to avoid reconstructing a graph for each target word, since the entire sentence can be encoded in the personalization vector. In Reference [8], a possible and more accurate alternative is also presented, called **Personalized PageRank word-to-word (PPRw2w)**, where a different customization vector is used for each word of a sentence. Although significantly less efficient in terms of time complexity, this approach guarantees the best accuracy, so it is considered state of the art in unsupervised WSD. In particular, there is research in this field for the Arabic language, which study the effect of CD in the process of disambiguation for **the information retrieval system (IRS)**.

The WSD method proposed in Reference [14] is based on the global and local context, where the correct sense is considered as the one that has a closer semantic similarity to both local and global context. In this method, the local context is defined by the neighborhood of an ambiguous word, and the global context is defined by the whole text. Their proposed approach provides an accuracy of 74%.

References [49, 50] propose to change the Lesk algorithm by using similarity measures like Wu, Palmer's, Harman, Okapi, and Croft, based on the distance between two nodes of the hierarchy and their position relative to the root. The Resnik measure, the Jiang and Conrath measure, the Lin measure and the Chodorow and Leacock measure, all aim to find the gloss corresponding to the meaning of the word to disambiguate for Arabic texts. This technique is tested using the dictionary "Al-Mujam al-Wasit"[7] with the corpus "Latif-Al Sulaiti."[8] According to the returned results, Leacock and Chodorow's measure is the most precise compared to Lesk, with a precision equal to 67% and 59%, respectively, unlike the other measures, which provide minimal results compared to Lesk.

In the other work cited in Reference [48], the authors proposed a hybrid approach for Arabic WSD combining an unsupervised and a knowledge-based method. Their experiments are based on both "Al-Mujam al-Wasit" and AWN. This approach achieves precision of 79% and 71% F-score measure.

In the work cited in Reference [10], the authors present multiple approaches for the problem of Arabic WSD utilizing the word embeddings with approaches such as GloVe and Word2vec. Their contribution in this work is to computationally obtain an embedding for each sense using AWN

[7] https://www.almaany.com/appendix.php.

[8] http://www.comp.leeds.ac.uk/eric/latifa/research.htm.

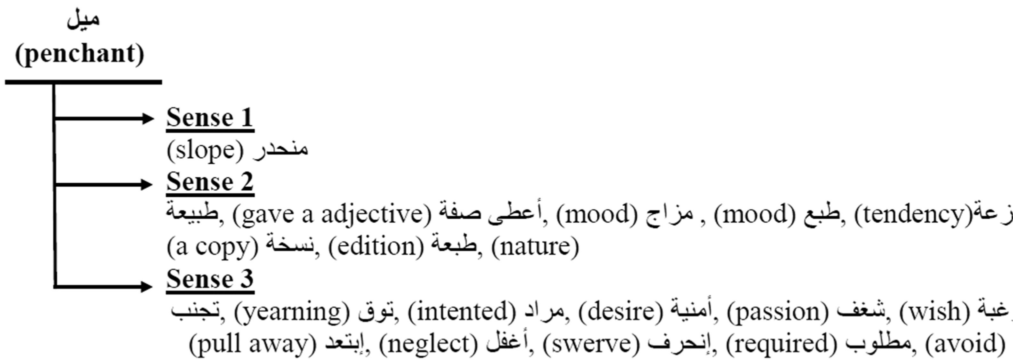

Fig. 1. The senses of (ميل "penchant") in AWN.

to overcome the problem of WSD. They used also Arabic stemming algorithms to compute word semantic similarity. By using Light stemmer 10 and GloVe, Reference [10] achieves the best result of similarity.

3 CONCEPTUAL DENSITY AND WORD SENSE DISAMBIGUATION

The relationship and distance between the concepts in the semantic resources are conceived as an important component of information in the process of disambiguation. This information can be used for calculating the CD.

The CD consists of calculating the distance between the senses of a given word and their context in WordNet. The algorithm consists of determining the shortest path in the sub-hierarchy extracted from WordNet.

According to Reference [5], the CD among concepts depends on the following:

- The shortest path that connects the concepts involved;
- The deepest concepts are the closest;
- The density of concepts in the hierarchy: concepts in a dense part of the hierarchy are relatively closer than those in a sparser region; and
- The closest concepts in the hierarchy.

Figure 1 shows the senses of "ميل (Penchant)" in AWN.

The example presented in Figure 1 shows the definitions of the word (ميل "penchant") extracted from the AWN. Each circle represents a sub-hierarchy containing the meaning of the word. Figure 1 shows that there are three different senses. The words in bold at the top of the sets represent the meaning of the word to disambiguate. The CD algorithm calculates and chooses the sense that has the highest value. In Figure 1, sense three will be chosen.

Let us define the CD as follows [5]:

$$CD(c, m) = \frac{\sum_{i=0}^{m-1} nhyp^i}{\sum_{i=0}^{h-1} nhyp^i}, \tag{1}$$

where c is the Synset at the top of the sub-hierarchy, m is the number of word senses falling a sub-hierarchy, h is the height of the sub-hierarchy, and *nhyp* is the average number of hyponyms for each node (Synset) in the sub-hierarchy. In the numerator, it counts the meanings of the ambiguous word with the words of their contexts in the sub-hierarchy, and in the divisor, it counts all the nodes of the sub-hierarchy [5]. The base formula considers the M number of relevant Synsets, divided by the total number nh of Synsets of the sub-hierarchy:

$$baseCD(M, nh) = \frac{M}{nh}. \tag{2}$$

3.1 Conceptual Density with Frequency

The addition of the notion of frequency in the CD formula is used for the first time by Reference [17]. This addition makes it possible to do the following:

- Solve the problem of the same values of CD for some senses;
- Avoid the case of failure of the algorithm of CD disambiguation; and
- Allow selecting a sense that has importance in WordNet. Let us define the CD as follows:

$$CD(M, nh, f) = M^{\alpha} \left(\frac{M}{nh}\right)^{\log f}, \tag{3}$$

where M is the number of relevant Synsets, α is a constant used to smooth the values of CD between 1 and the total number of senses in WordNet, f is an integer representing the frequency of the sense in WordNet (1 represents the most frequent, 2 the second most frequent, etc.). This formula makes it possible to give a density equal to 1 to the most frequent sense, if the CD algorithm selects the less frequent meaning, that is, the density exceeds 1. This formula favors the sub-hierarchies with a greater number relevant Synsets [17].

4 THE DISAMBIGUATION ALGORITHM USING CONCEPTUAL DENSITY

The algorithm has as input the graphic word (WA) and as output the best sense. First, the algorithm is represented in a tree extracted from AWN of the graphical word to disambiguate with the Synset S. Then, the algorithm computes the CD of each concept in WordNet according to the senses it contains in its sub-hierarchy (CDi). The CD is computed by the formula cited in 2 and 3. Then, the algorithm selects the highest CD (Best_Score) and selects the Synset below it as the correct sense for the respective words Best_Synset. If there is more than one Synset with the same Best_Score, then the algorithm considers it as a failure, and it cannot remove the ambiguity of the concerned word. The algorithm continues to disambiguate all the words of texts to produce a better information representation.

This algorithm treats one word at a time from the beginning of the document toward its end, disambiguating the word in the middle of the window in each step, considering the other words in the window as context [5].

The algorithm proceeds as follows:

ALGORITHM 1: Disambiguation with Conceptual Density

(1) **Input:** WA, AWN
(2) **Output:** Best_Synset
(3) **Begin**
(4) P(WA) = $C(WA) \cup WA$
(5) Best_Score = 0
(6) Best_Synset = ""
(7) S = Candidates_Synset_Arabic_WordNet(WA)
(8) **If** (Card(S)>1) **Then**
(9) TreeWA = Extract_sub-hierarchy(WA)
(10) **For each** ($Si \in S$) **Do**
(11) TreeSi = Extract_sub-hierarchy_From_TreeWA(Si)

```
            nh = Card (TreeSi)
            M = Card(P(WA) ∩ TreeSi)
            CDi = CD(M, nh)
            If (Best_Score<CDi) Then
                Best_Score = CDi
                Best_Synset = Si
            End If
        End For
    End If
End
```

Where:

- **WA:** The graphical word to disambiguate.
- **C(WA):** The context of WA.
- **P(WA):** WA with its context.
- **Best_Score:** The highest score of synonym.
- **Best_Synset:** The most appropriate sense of WA.
- **Candidates_Synset_Arabic_WordNet(WA):** This function extracts the Synsets, which contains WA.
- **Card(S):** The cardinality of S.
- **Extract_sub-hierarchy(WA):** This function extract the sub-hierarchy of WA, this is means, extract the Synsets related to the Synsets extrated by Candidates_Synset_Arabic_WordNet.
- **Extract_sub-hierarchy_From_TreeWA(Si):** This function extract the sub-hierarchy of Si from TreeWA.
- **Card (TreeSi):** This function calculate the number of Synsets in the sub-hierarchy Si.
- **CD(M, nh):** This function calculate the CD for the sub-hierarchy Si by using Equation (2) or Equation (3).

4.1 Example

The example of the adopted approach is given in below. In the input, we have the following Arabic text:

يعد إنتهاك وخرق المعاهدات وتجاوز القوانين أو تغييرها سببا لإلغاء الإتفاقيات المبرمة

The violation and breach of treaties and overriding or changing of laws is a reason to cancel the agreements concluded.

We want to remove the ambiguity of the word "إنتهاك" (violation), which has the following attributes:

- Three possible senses in AWN: {"ذنب" (sin), "مخالفة" (infraction), "غزو (invasion)}; and
- The terms of Context: {"خرق" (breach), "معاهدات" (treaties), "تجاوز" (override), "قوانين" (laws) "تغيير" (change), "سبب" (reason), "إلغاء" (cancel), "إتفاقيات" (agreements) and "مبرمة" (concluded)}.

Figure 2 shows the sub-hierarchy of the word "إنتهاك" (violation) with its context.

The areas of sub-hierarchies are numerated from 1 to 3; the root of sub-hierarchies are the darker nodes; the nodes corresponding to the synsets of the word to be disambiguated are drawn with a

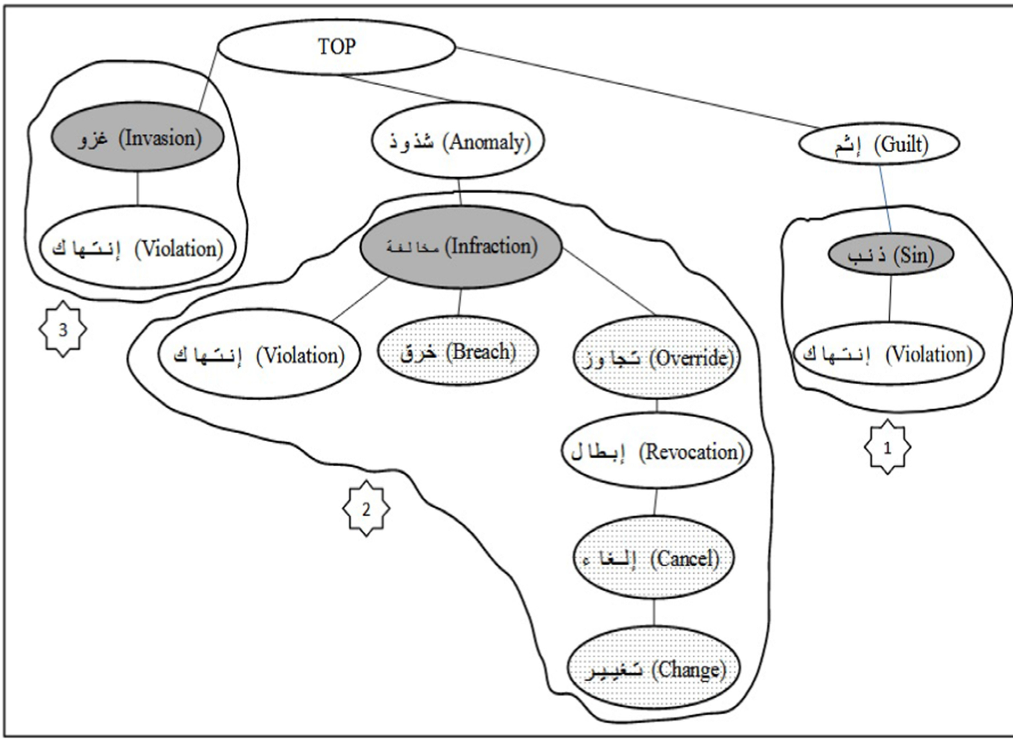

Fig. 2. The sub-hierarchy of the word “إنتهاك” (violation) with their context extracted from AWN.

thicker border, while the context words are drawn with a dotted background. Three sub-hierarchies have been identified, one for each sense of “إنتهاك” (violation), as follows:

- sub-hierarchy_1 = {“إنتهاك” (violation), “غزو” (invasion)};
- sub-hierarchy_2 = {“إنتهاك” (violation), “تغيير” (change), “مخالفة” (infraction), “خرق” (breach), “إلغاء” (cancel), “إبطال” (revocation), “تجاوز” (override)}; and
- sub-hierarchy_3 = {“إنتهاك” (violation), “ذنب” (sin)}.

Context words falling outside of these sub-hierarchies is not considered.

The results of the CD algorithm for each sub-hierarchy are as follows:

- M1 = M3 = 1;
- nh1 = nh3 = 2;
- M2 = 5; and
- nh2 = 7.

where Mi and nhi indicate the M and nh values for the *i*th sense, respectively:

- CD_sub1 = CD_sub3 = 1/2 = 0.50; and
- CD_sub2 = 5/7 = 0.71.

Therefore, the second one, “مخالفة”, is selected and the second sense is assigned to “إنتهاك” (violation).

The results of the CD with frequency for each sub-hierarchy are as follows:

- M1 = M3 = 1;
- nh1 = nh3 = 2;
- M2 = 5;
- nh2 = 7; and
- f_S1 = 1, f_S2 = 2, f_S3 = 3.

where Mi and nhi indicate the M and nh values for the *i*th sense and f_S indicate the frequency of sense, respectively:

- $CD\_sub1 = 1^{0.1} * (1/2)^{log1} = 1.000$,
- $CD\_sub2 = 5^{0.1} * (5/7)^{log2} = 1.061$,
- $CD\_sub3 = 1^{0.1} * (1/2)^{log3} = 0.718$,

Therefore, the second one, "مخالفة", is selected and the second sense is assigned to "إنتهاك" (violation).

5 THE DISAMBIGUATION ALGORITHM USING PAGERANK

Through the following formula, the PageRank method makes it possible to classify the nodes (senses) of a graph to find the best one [8]:

$$Pr = (1 - \alpha) + \alpha * MPr, \tag{4}$$

where α is a damping factor between 0 and 1 (the factor is usually fixed at 0.85 [8]) and M is a matrix of the weights of arcs initialized to $1/N$ if the arc (i, j) exists; otherwise, the weight of the arc equals 0.

The PageRank algorithm [15] is used for classifying nodes of a graph according to their relative structural importance. This algorithm has been proposed for ranking web pages for the IR. Reference [29] adapted this approach for an application in WSD.

With PageRank, whenever a link from i to j exists on a graph, a vote from node i to node j is produced, and the rank of node j increases. In addition, the strength of the vote from i to j depends on the rank of node i. The more important node i is, the stronger its vote.

The algorithm initializes the ranks of the nodes with a fixed value (usually $1/N$ for a graph with N vertices) and iterates until the algorithm traverses the whole graph. The algorithm selects the best sense at the highest rank. The constructed graph makes it possible to disambiguate the complete text by entering it. The algorithm proceeds as follows:

ALGORITHM 2: Disambiguation with PageRank

```
Input: WA, AWN
Output: Best_Synset
Begin
    Best_Score = 0
    Best_Synset = ""
    S = Candidates_Synset_Arabic_WordNet(WA)
    If (Card(S)>1) Then
        GraphWA = Extract_Graph(WA)
        N = card (GraphWA)
        For each Si ∈ GraphWA Do
            Pr_Si = 1/N
        End For
        For each Si ∈ S Do
```

```
            Pr_Si = (1 − α) + α ∗ Pr_Si ∗ Nb_Predecessor_of_Si
            If (Best_Score<Pr_Si) Then
                Best_Score = Pr_Si
                Best_Synset = Si
            End If
        End For
    End If
End
```

Where:

- **S:** The Synsets of WA.
- **Pr_Si:** This value is the rank of synsets.
- **Candidates_Synset_Arabic_WordNet(WA):** This function extracts the Synsets, which contains WA.
- **Extract_Graph(WA):** This function extract the graph contains WA, this is means, this function extract all Synsets related to the Synsets Extrated by Candidates_Synset_Arabic_WordNet.
- **α:** a damping factor between 0 and 1. The factor is usually fixed at 0.85 (see page 10).
- **Nb_Predecessor_of_Si:** This value is the number of the Synset predecessor.

The algorithm has in input a graphic word **WA** and as output the best sense of this word. First, the algorithm represents the sentence that contains the graphical word **WA** in a graph **GraphWA** extracted from **AWN**. Then for each sense Si of WA; the rank of sense depends on the rank of each synsets predecessors.

The rank of sense Si is computed by the formula cited in 4. Then the algorithm selects the highest rank **Best_Score** and selects the Synset below it as the correct senses for the respective words **Best_Synset**.

5.1 Example

An example of the adopted approach is given bellow. In the input, we have the Arabic text:

يعد إنتهاك وخرق المعاهدات وتجاوز القوانين أو تغييرها سبب لإلغاء الإتفاقيات المبرمة

The violation and breach of treaties and overriding or changing of laws is a reason to cancel the agreements concluded.

We want to remove the ambiguity of the word "إنتهاك" (violation), which has the following attributes:

- Three possible senses in AWN: {"ذنب" (sin), "مخالفة" (infraction),"غزو" (invasion)}; and
- The terms of Context: {"خرق" (breach), "معاهدات" (treaties), "تجاوز" (override), "قوانين" (laws) "تغيير" (change), "سبب" (reason), "إلغاء" (cancel), "إتفاقيات" (agreements) and "مبرمة" (concluded)}.

Figure 3 shows the graph of the word "إنتهاك" (violation) with its senses in the sentence.

Let:

- S = {"ذنب" (sin), "مخالفة" (infraction),"غزو" (invasion)}
- Card(S) = 3

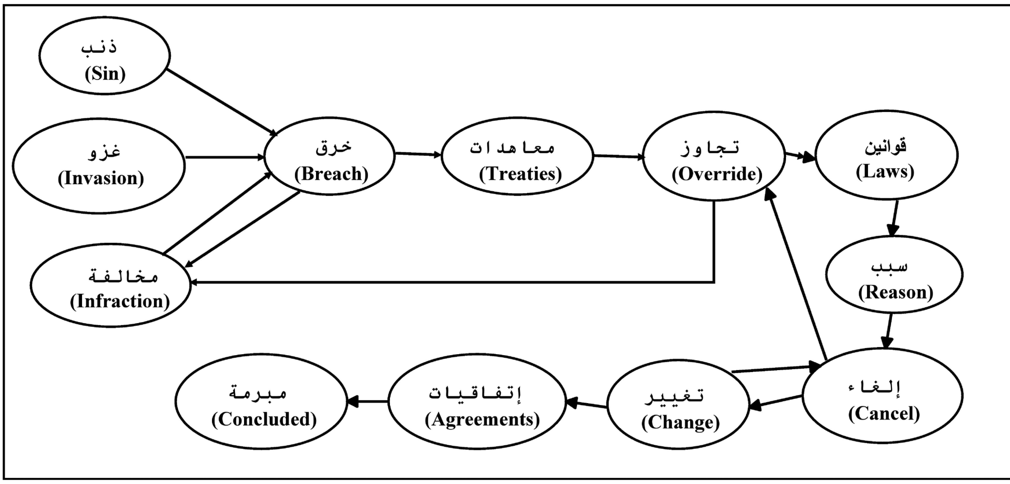

Fig. 3. The graph of the word "إنتهاك" (violation) with the terms of their context extracted from AWN.

- GraphWA = {"ذنب" (sin), "مخالفة" (infraction), "غزو" (invasion), "خرق" (breach), "معاهدات" (treaties), "تجاوز" (override), "قوانين" (laws) "تغيير" (change), "سبب" (reason), "إلغاء" (cancel), "إتفاقيات" (agreements) and "مبرمة" (concluded)}
- α = 0.85 (this value is recommanded by Reference [8], see page 10)
- N = card (GraphWA) = 12 *//There are 12 synsets in the graph contains WA extracted from AWN, and **Pr** of each synset in the **GraphWA** is initialised to 1/12.*

The PageRank algorithm calculates for each Si its Pr_Si (Pr_"ذنب", Pr_"مخالفة" and Pr_"غزو") and according to this value it deduces the scores. The values are as follows:

- Pr_"ذنب" = (1 – 0.85) + 0.85 * 0 = 0.15. *//Synset "ذنب" have no predecessors.*
- Pr_"غزو" = (1 – 0.85) + 0.85 * 0 = 0.15. *//Synset "غزو" have no predecessors.*
- Pr_"مخالفة"= (1 – 0.85) + 0.85 * (Pr_"خرق"+Pr_"تجاوز") = 0.15 + 0.85 * (1/12+1/12) = 0.29 *//Synset "مخالفة" have 2 predecessors.*

Therefore, the second one, "مخالفة," is selected and the second sense is assigned to "إنتهاك" (violation).

6 EVALUATION OF THE USE OF PAGERANK AND CONCEPTUAL DENSITY

To measure the impact of word disambiguation in the IRS, the main goal of the experiments is to evaluate the IRS performance.

6.1 Data Set

Two corpora are used for the evaluation:

Corpus 1 (C1): this corpus is developed in our laboratory for which we do not have the necessary permission to share it on the net. All the documents in this corpus are downloaded from internet. It contains 22,000 Arabic documents from different fields as shown in Table 1. In this Arabic dataset, each document was saved in a separate file within the corresponding category's directory. For the interrogation of this corpus, we have used 100 queries, each query consists of a single word.These queries cover practically the majority of the categories existing in this

Table 1. Arabic Text Corpus

Categories	Number of Documents
Astronomy	557
Law	944
Economy	3,102
Education	3,608
History	3,233
Stories	726
Recipes	2,373
Religion	3,171
Health	2,296
Sport	2,419

corpus. For each category, we used seven queries. As an example of query: "فضاء" (Space) for the astronomy category, "قصص" (Stories) for the Stories category, "وصفة" (Recipe) for the Recipes category, " إثم"(Sin), "دين" (Religion), for the religion category, "مرض" (Disease), "صحة" (Health)) for the health category, and so on.

Corpus 2 (C2): Essex Arabic Summaries Corpus (EASC[9]). It contains 153 Arabic documents from different fields as shown in Table 2. In this Arabic dataset, each document was saved in a separate file within the corresponding category's directory. For the interrogation of this corpus, we have used 50 queries, each query consists of a single word. These queries cover practically the majority of the categories existing in this corpus. For each category, we used seven queries. As an example of query: ("تأليف" (Composition), "مؤلف" (Composer), "صوت" (Voice)) for the category of Art and Music, ("دراسة" (Study), " علم" (Science), " تعليم" (Education)) for the education category, ("محيط" (Environment), "سطح" (Science), "مساحة" (Surface)) for the environment category, ("استثمار" (Investment), "نقود" (Money), "مالية" (Finance)) for the finance category, ("ضغط" (Tension), "مرض" (Disease), "صحة" (Health)) for the health category, "فضاء" (Space) for the astronomy category, "قصص" (Stories) for the Stories category, "وصفة" (Recipe) for the Recipes category, and so on.

6.2 Evaluation Metrics

The evaluation is based on metrics include: precision, recall, and F-score. The idea is to divide the documents returned by the IRS to relevant documents and irrelevant documents, this can be made clear by the confusion matrix (see Table 3) and the following formulas [1, 38], where

- TP is the relevant documents retrieved,
- FN is the relevant documents not retrieved,
- FP is the irrelevant documents retrieved,
- TN is the irrelevant documents not retrieved:

$$Precision = \frac{TP}{TP + FP}, \tag{5}$$

[9]https://sourceforge.net/projects/easc-corpus/.

Table 2. EASC's Arabic Text Corpus

Categories	Number of Documents
Art and Music	10
Education	07
Environment	33
Finance	17
Health	17
Politics	21
Religion	08
Science and Technology	16
Sports	10
Tourism	14

Table 3. Confusion Matrix

	Relevant document	Irrelevant document
Retrieved	True Positive (TP)	False Positive (FP)
Not Retrieved	False Negative (FN)	True Negative (TN)

$$Recall = \frac{TP}{TP + FN}, \tag{6}$$

$$F - score = \frac{2 * Precision * Recall}{Precision + Recall}. \tag{7}$$

We have also used the precision/recall curve. The traditional way of doing this is the 11-point interpolated average precision. For each query, the interpolated precision is measured at the 11 recall levels of $0.0, 0.1, 0.2, \ldots, 1.0$. Then, for each recall level, we calculate the arithmetic mean of the interpolated precision at that recall level for each query in the test collection. The 11 levels of the recall" allow us to have precision values at 0, 0.1, 0.2, 0.9 and 1 as recall values. It also allows us to plot the recall curve as a function of precision.

6.3 Experimentation and Results

To evaluate the use of PageRank and the CD in WSD for Arabic IRS, four different types of searches were conducted. We will study them separately to measure the contribution of each type in improving the performance of the IRS.

In this setup, the process of disambiguation is used in **semantic indexation (SI)** for IRS. SI is a crucial process in the search operation, it is based on the sense of words, instead of single words, to represent the documents and the queries [31].

By using C1 and C2, four different types of searches were conducted:

- Search without WSD (R0),
- Search with WSD by using CD (R1),
- Search with WSD by using CDF (R3),
- Search with WSD by using PageRank (R4).

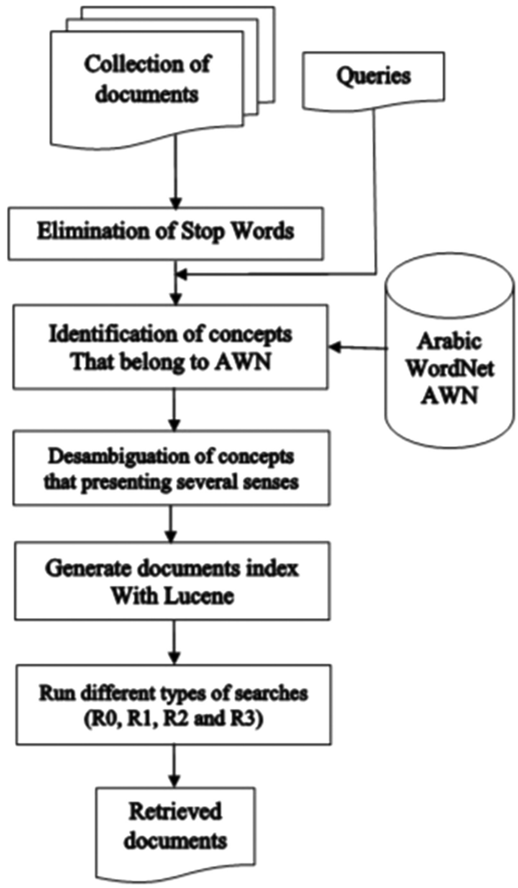

Fig. 4. The different steps of our experimentation.

We will study them separately to measure the contribution of each type in improving the performance of the IRS.

For the implementation of our experimental setting, we have used the Lucene[10] API and the java language.

In the input of our system, we have the texts and the queries and in the output, we have the texts retrieved by these queries, and inside this process, we find the operation of disambiguation. Figure 4 shows the different steps of our experimentation.

Table 4 shows the results of recall, precision, and f-score of R0, R1, R2, and R3. This results are for C1 and C2. The results presented in Table 4 show that R3 achieved the best performance. Specifically, the method based on Search with WSD by using PageRank present the best values of recall, precision, and F-score for the two Corpus (C1 and C2).

Figures 5 and 6 show the curve of the 11 levels of the recall of R0, R1, R2, and R3, they clearly show the superiority of R3. The 11 levels of the recall allows us to have precision values at 0, 0.1, 0.2, 0.9, and 1 as recall values. It also allows us to plot the recall curve as a function of precision.

The analysis of the obtained results of the four systems R0, R1, R2, and R3 makes it possible to deduce that the performance of the IR systems improved during the disambiguation of the

[10] http://www.apache.org/dyn/closer.cgi/lucene/java/.

Table 4. Results of Recall, Precision, and F-score of R0, R1, R2, and R3 for C1 and C2

		Recall	Precision	F-score
C1	R0	0,31	0,49	0,38
	R1	0,35	0,58	0,42
	R2	0,39	0,63	0,44
	R3	**0,45**	**0,69**	**0,51**
C2	R0	0,34	0,59	0,41
	R1	0,39	0,61	0,44
	R2	0,40	0,65	0,47
	R3	**0,47**	**0,71**	**0,52**

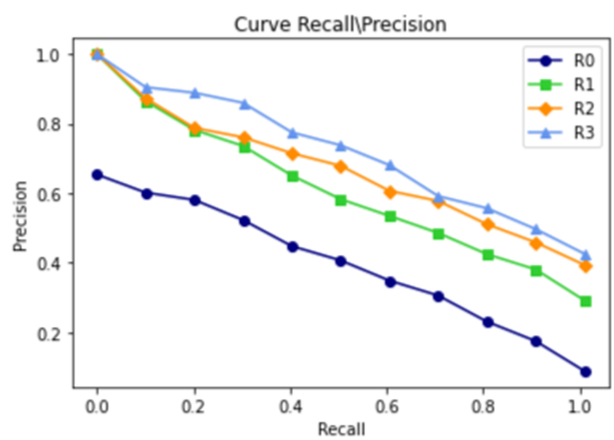

Fig. 5. Curves recall/precision according to the type of search for C1.

words (R1, R2, R3). The comparison of the two systems (R1, R2) shows that the variant of the CD algorithm with addition of the frequency R2 makes it more efficient compared to the basic CD algorithm. Therefore, the optimal algorithm is the one used in R2.

The comparison of the systems (R1, R2, R3) shows that the PageRank algorithm R3 makes it more efficient compared to the CD algorithm. Thus, the optimal algorithm is the one used in R3.

The comparison of the disambiguation algorithm through PageRank and CD with the Lesk algorithm [2] for IR in Arabic text, shows that PageRank and CD are better in term of precision and give stronger search results for search compared with Lesk. It should also be noted that CD is faster than Lesk, while PageRank is slower. This is due to the principle of the CD algorithm that locates the ambiguous words and their contexts in AWN. It can then deduce all the relevant meaning of the ambiguous words at the same time, unlike the Lesk algorithm, which is very heavy, because it calculates the overlap of a word and its contexts with the glosses found in AWN. The PageRank algorithm is slower, because the rank of each synonym is dependent on its predecessor, and it will be especially heavy for large graphs.

The absence of glosses in the AWN and the size of the context impact the quality of the results of the Lesk algorithm, whereas the PageRank and CD algorithms are more efficient in the choice of the best sense.

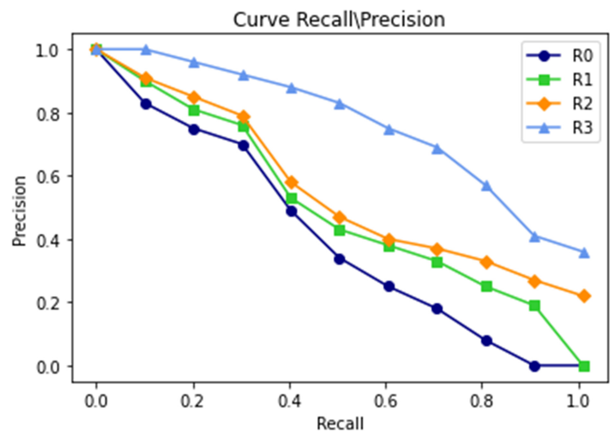

Fig. 6. Curves recall/precision according to the type of search for C2.

7 CONCLUSION

This study was about the evaluation of WSD by graph-based approach (PageRank and CD) used for Arabic texts in IRS. The experiments with the AWN for PageRank, CD, and CD with frequency for WSD in IRS allows for improving their precision. Two medium-sized corpora (about 630,914 distinct words) in different fields were used to evaluate the results obtained by the interrogation of these corpora using a set of queries. The lexical base of AWN was exploited in an IRS to index the collection of documents and the user query. Our experimentation showed us that disambiguation with PageRank and CD significantly enhances the quality of an Arabic IRS, with further advantage to the PageRank method compared to the CD method.

The results proved that the Random Walk can enhance the performance of information retrieval system by achieving a mean improvement of 13%, 16%, and 12% in terms of recall, precision, and F-score.

In future work, we will conduct experiments with approaches based on machine learning and deep learning for the disambiguation of Arabic texts, which will enable us to use Elmo [35], Bert [21], word2vec, and GloVe for extracting the best word representation in their context for Arabic texts.

REFERENCES

[1] M. A. Abderrahim. 2016. Exploitation des Ontologies dans les Systèmes de recherche d'informations Arabes. Thése de doctorat, Universitè de Tlemcen, Algerie.

[2] M. A. Abderrahim, M. Dib, M. A. Abderrahim, and M. A. Chikh. 2016. Semantic indexing of Arabic texts for information retrieval system. *Int. J. Speech Technol.* 19, 2 (June 2016), 229–236. https://doi.org/10.1007/s10772-015-9307-3

[3] E. Agirre, X. Arregi, X. Artola, A. Díaz de Ilarraza, and K. Sarasola. 1994. *Conceptual distance and automatic spelling correction.* Technical Report. Retrieved from http://www.researchgate.net/publication/2273948_Conceptual_Distance_and_Automatic_Spelling_Correction/file/79e4150b4d909428bc.pdf%5Cnhttp://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.43.9218.

[4] E. Agirre, O. López de Lacalle, and A. Soroa. 2014. Random walks for knowledge-based word sense disambiguation. *Comput. Linguist.* 40, 1 (Mar. 2014), 57–84. https://doi.org/10.1162/COLI_a_00164

[5] E. Agirre and G. Rigau. 1995. A proposal for word sense disambiguation using conceptual distance. Technical Report. https://doi.org/10.1075/cilt.136.16agi

[6] E. Agirre and G. Rigau. 1996. An Experiment in Word Sense Disambiguation of the Brown Corpus Using WordNet. Technical Report.

[7] E. Agirre and A. Soroa. 2008. Using the Multilingual Central Repository for Graph-Based Word Sense Disambiguation. Technical Report. 1388–1392. Retrieved from http://nipadio.lsi.upc.es/nlp/meaning.

[8] E. Agirre and A. Soroa. 2009. Personalizing PageRank for word sense disambiguation. 33–41. Retrieved from https://dl.acm.org/citation.cfm?id=1609070.

[9] R. Al-Shalabi, G. Kanaan, M. Yaseen, B. AlSarayreh, and N. Al-Naji. 2009. Arabic query expansion using interactive word sense disambiguation. In *Proceedings of the 2nd International Conference in Arabic Language Resources and Tools*.

[10] A. Alkhatlan, J. Kalita, and A. Alhaddad. 2018. Word sense disambiguation for arabic exploiting arabic WordNet and word embedding. *Procedia Comput. Sci.* 142 (2018), 50–60.

[11] S. Banerjee and T. Pedersen. 2003. Extended gloss overlaps as a measure of semantic relatedness. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI'03)*. 805–810.

[12] P. Basile, A. Caputo, and G. Semeraro. 2014. An Enhanced Lesk Word Sense Disambiguation Algorithm through a Distributional Semantic Model. Technical Report. 1591–1600. Retrieved from https://www.aclweb.org/anthology/C14-1151.

[13] F. Boubekeur, M. Boughanem, L. Tamine, and M. Daoud. 2010. De l'utilisation deWordNet pour l'indexation conceptuelle des documents. In *le 13 ème Colloque International sur le Document Electronique (CIDE 13), 16-17 Décembre 2010, INHA, Paris. France.* https://lorexplor.istex.fr/Wicri/Ticri/CIDE/fr/images/d/db/CIDE_%282010%29_Boubekeur.pdf.

[14] N. Bouhriz, F. Benabbou, and H. Ben Lahmar. 2016. Word sense disambiguation approach for arabic text. *Int. J. Adv. Comput. Sci. Appl.* 7, 4 (2016), 381–385.

[15] S. Brin and L. Page. 1998. The Anatomy of a Large-Scale Hypertextual Web Search Engine The Anatomy of a Search Engine. Technical Report. 107–117. https://doi.org/10.1016/S0169-7552(98)00110-X

[16] D. Buscaldi and P. Rosso. 2008. A conceptual density based approach for the disambiguation of toponyms. *Int. J. Geogr. Info. Sci.* 22, 3 (Mar. 2008), 301–313. https://doi.org/10.1080/13658810701626251

[17] D. Buscaldi, P. Rosso, and F. Masulli. 2004. Integrating conceptual density with WordNet Domains and CALD glosses for noun sense disambiguation. In *Advances in Natural Language Processing*. Springer, Berlin, 183–194. https://doi.org/10.1007/978-3-540-30228-5_17

[18] J. Camacho-Collados, M. T. Pilehvar, and R. Navigli. 2016. Nasari: Integrating explicit knowledge and corpus statistics for a multilingual representation of concepts and entities. *Artific. Intell.* 240 (Nov. 2016), 36–64. https://doi.org/10.1016/J.ARTINT.2016.07.005

[19] W. Che, Y. Liu, Y. Wang, B. Zheng, and T. Liu. 2018. Towards better UD parsing: Deep contextualized word embeddings, ensemble, and treebank concatenation. In *Proceedings of the CoNLL Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies.* Association for Computational Linguistics, Brussels, Belgium, 55–64. Retrieved from http://www.aclweb.org/anthology/K18-2005.

[20] J. Cowie, J. Guthrie, and L. Guthrie. 1992. Lexical disambiguation using simulated annealing. In *Proceedings of the 14th conference on Computational linguistics*, Vol. 1. Association for Computational Linguistics, 359–365. https://doi.org/10.3115/992066.992125

[21] J. Devlin, M. Chang, K. Lee, and K. Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1.* Association for Computational Linguistics, 4171–4186.

[22] A. Farghaly and K. Shaalan. 2009. Arabic natural language processing: Challenges and solutions. *ACM Trans. Asian Lang. Inform. Process* (2009).

[23] A. Gliozzo, B. Magnini, and C. Strapparava. 2004. *Unsupervised Domain Relevance Estimation for Word Sense Disambiguation.* Technical Report. 380–387. Retrieved from http://wndomains.itc.it.

[24] M. Hadni, S. E. A. Ouatik, and A. Lachkar. 2016. Word sense disambiguation for arabic text categorization. *Int. Arab J. Inf. Technol. 13(1A)* (2016), 215–222.

[25] M. E. Lesk. 1986. Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from a nice cream cone. In *Proceedings of the SIGDOC Conference*.

[26] B. Magnini and G. Cavaglia. 2000. Integrating subject field codes into WordNet. *Proceedings of the 2nd International Conference on Language Resources and Evaluation Theoretical Aspects of Computer Software.* 1413–1418.

[27] B. Magnini, C. Strapparava, G. Pezzulo, and A. Gliozzo. 2002. The role of domain information in Word Sense Disambiguation. *Natural Lang. Eng.* 8, 4 (Dec. 2002), 359–373. https://doi.org/10.1017/S1351324902003029

[28] M. B. Menai. 2014. Word sense disambiguation using an evolutionary approach. *Informatica* 38, 2 (2014), 155–169.

[29] R. Mihalcea, T. Chklovski, and A. Kilgarriff. 2004. *The SENSEVAL-3 English Lexical Sample Task.* Technical Report. 25–28. Retrieved from http://digital.library.unt.edu/ark:/67531/metadc30963/.

[30] R. Mihalcea and D. I. Moldovan. 1999. A method for word sense disambiguation of unrestricted text. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics.* Association for Computational Linguistics. 152–158. https://doi.org/10.3115/1034678.1034709

[31] R. Mihalcea and D. I. Moldovan. 2000. Semantic indexing using WordNet senses. In *Proceedings of the ACL Workshop on IR & NLP.* 35–45. Retrieved from http://www.seas.smu.edu/~rada/papers/acl00.nlp_ir.ps.gz.

[32] S. Mohammad and G. Hirst. 2006. *Determining Word Sense Dominance Using a Thesaurus*. Technical Report. 121–128. Retrieved from https://www.aclweb.org/anthology/E06-1016.

[33] A. Moro, A. Raganato, and R. Navigli. 2014. *Entity Linking meets Word Sense Disambiguation*. Technical Report. 231–244. https://doi.org/:10.1371/journal.pone.0098221

[34] R. Navigli and M. Lapata. 2007. Graph connectivity measures for unsupervised word sense disambiguation. Technical Report. 1683–1688. Retrieved from https://www.research.ed.ac.uk/portal/files/24353052/IJCAI07{_}272.pdf.

[35] M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, volume 1*. Association for Computational Linguistics, 2227–2237.

[36] S. P. Ponzetto and R. Navigli. 2010. Knowledge-rich Word Sense Disambiguation rivaling supervised systems. 1522–1531. Retrieved from https://dl.acm.org/citation.cfm?id=1858835.

[37] G. Rigau, J. Atserias, and E. Agirre. 1997. Combining unsupervised lexical knowledge methods for word sense disambiguation. In *Proceedings of the 35th annual meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 48–55. https://doi.org/10.3115/976909.979624

[38] C. J. Van Rijsbergen. 1979. Information retrieval. *J. Amer. Soc. Info. Sci.* 30, 6 (1979), 374–375.

[39] P. Rosso, F. Masulli, D. Buscaldi, F. Pla, and A. Molina. 2003. Automatic noun sense disambiguation. In *Lecture Notes Computer Science*. Springer, Berlin, 273–276. https://doi.org/10.1007/3-540-36456-0_27

[40] H. Schutze. 1998. Automatic word sense discrimination. *Comput. Linguist.* 24, 1 (1998), 97–123.

[41] R. Sinha and R. Mihalcea. 2007. Unsupervised graph-based word sense disambiguation using measures of word semantic similarity. Technical Report. 363–369. https://doi.org/10.1109/ICSC.2007.87

[42] M. Sussna. 1993. Word sense disambiguation for free-text indexing using a massive semantic network. Technical Report. 67–74. https://doi.org/10.1145/170088.170106

[43] R. Tripodi and M. Pelillo. 2017. A game-theoretic approach to word sense disambiguation. *Comput. Linguist.* 43, 1 (2017), 31–70. https://doi.org/10.1162/COLI_a_00274

[44] E. Voorhees and M. Ellen. 1993. Using WordNet to disambiguate word senses for text retrieval. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'93)*. ACM Press, 171–180. https://doi.org/10.1145/160688.160715

[45] D. Weissenborn, L. Hennig, F. Xu, and H. Uszkoreit. 2015. *Multi-Objective Optimization for the Joint Disambiguation of Nouns and Named Entities*. Technical Report. 596–605. https://doi.org/10.3115/v1/p15-1058

[46] Y. Wilks, D. Fass, C. Guo, J. E. McDonald, T. Plate, and B. M. Slator. 1990. Providing machine tractable dictionary tools. *Mach. Translat.* 5, 2 (June 1990), 99–154. https://doi.org/10.1007/BF00393758

[47] D. Yarowsky. 1992. Word-sense disambiguation using statistical models of Roget's categories trained on large corpora. Technical Report. 454–460. https://doi.org/10.3115/992133.992140

[48] A. Zouaghi, L. Marhbène, and M. Zrigui. 2012. A hybrid approach for Arabic word sense disambiguation. *Int. J. Comput. Process. Lang.* 24, 2 (2012), 133–151.

[49] A. Zouaghi, L. Merhbene, and M. Zrigui. 2011. Word Sense disambiguation for Arabic language using the variants of the Lesk algorithm. Technical Report. 561–567. Retrieved from http://www.lidi.info.unlp.edu.ar/WorldComp2011-Mirror/ICA4686.pdf.

[50] A. Zouaghi, M. Zrigui, G. Antoniadis, and L. Merhbene. 2012. Contribution to semantic analysis of arabic language. In *Advances in Artificial Intelligence*. Springer 1–8. https://doi.org/10.1155/2012/620461

Received January 2020; revised May 2021; accepted October 2021