Word sense disambiguation of Arabic language with Word Embeddings as part of the Creation of a Historical Dictionary

Rim Laatar, Chafik Aloulou, and Lamia Hadrich Bilguith

ANLP Research Group, MIR@CL Lab,
Faculty of Economics and Management,
University of Sfax, Tunisia.

rimlaatar@yahoo.fr
{chafik.aloulou, l.belguith}@fsegs.rnu.tn

Abstract. A historical dictionary is a dictionary which deals with the detailed history of words since their first appearance in language as well as the evolution of their meaning and use throughout history. To create such a dictionary, we are bound to follow many steps. As part of these steps, the extracting of the appropriate meaning of a given word occurring in a given context, named also word sense disambiguation (WSD). This article proposes a word embedding based method to solve the problem of WSD. The main idea is to exploit vectors as word representations in a multidimensional space in order to capture the semantic and syntactic properties of words. The experiments show that the proposed system achieved an accuracy of 78%.

1 Introduction

Linguists state that the major goal of the historical dictionary of Arabic resides in encompassing the entire Arabic lexicon throughout its history by presenting every Arabic word in its morphological, semantic and contextual development from its first appearance in written texts to the present. It shows definitions in an order that the meaning of the word being used allows the reader to get an approximate meaning of the time period in which a particular word has been in use.

The historical dictionary of Arabic is very significant because it not only remedies the present lack of an all-encompassing, historical dictionary for Arabic speakers, but also serves to preserve the Arab nation's common linguistic and intellectual legacy. Hence, the chief goal of the dictionary is to safeguard the riches of Arabic cultural heritage.

According to [1], we assume that the creation of a historical dictionary can benefit from automatic processing tools such as semantic analysis.

One of the possible steps to create a historical dictionary is by extracting the appropriate sense of a given word occurring in a given context and by recording the transformation of each word's meaning.

To our knowledge, it appears that there is no research addressing the issue of disambiguating Arabic words to create a historical dictionary of the Arabic language. In fact, WSD is the problem of identifying the meaning of a word within a specific context.

In this work, we will present a workable method for Arabic WSD based on word embedding. More particularly, the proposed system uses the Arabic dictionary to select word senses. Then, the sense attributed to an ambiguous word is the one that possesses the closest semantic proximity to the local context. Our method consists of first training word vectors from the corpus using Mikolov's Skip-Gram model [3], followed by representing the context of a word to be disambiguated and all the senses as a vector in a multidimensional space. Then, WSD is done by simple calculation of cosine similarity as a metric for comparing the similarity of the context vector with the target word sense vectors, the sense of the highest similarity being allocated as the disambiguated sense.

The rest of this article is organized as follows: the second section describes the main used approaches for WSD, the third one presents our proposed WSD method based on word embedding and the fourth one describes the experimental results of this study. Finally, our conclusion and some future works are drawn in Section five.

2 Main used approaches

WSD is a fundamental task in Natural Language Processing (NLP). The aim of WSD is to assign the correct meaning or the sense of a word in a given context. There are three main approaches to WSD: knowledge based approach, supervised approach and unsupervised approach.

2.1 Knowledge based approach

Knowledge based approaches are based on different knowledge sources as dictionaries, thesauruses and lexicons. This technique is applied to make use of one or more sources of knowledge to associate the most appropriate senses with words in context. Some of them are based on the calculation of the word overlap between the sense definitions of two or more target words [4]. This approach is named gloss overlap or the Lesk algorithm [5]. Yet some others have exploited a number of measures of semantic similarity based on the network of semantic connections between word senses in Wordnet.

2.2 Supervised based approach

They used an annotated training corpus for inducing a classifier from manually sense-annotated data sets. Usually, the classifier is concerned with a single word and performs a classification task in order to assign the appropriate sense to each instance of that word [4]. For the supervised methods, we can cite: the decision lists, neural networks and naive bayes algorithm.

2.3 Unsupervised approach

These methods are based on unlabeled corpora and do not exploit any manually sense-tagged corpus to provide a sense choice for a word in context. These approaches to WSD hinge upon the idea that the same sense of a word has similar neighboring words. They are able to induce word senses from input text by clustering word occurrences and then classifying new occurrences into the induced clusters [4]. They are divided into methods based on context clustering, word clustering and co-occurrence graphs. The first one represents each occurrence of a target word in a corpus as a context vector. Then, the vectors are clustered into groups, each identifying a sense of the target word. The second one cluster words which are semantically similar and can thus convey a specific meaning.

Finally, the last one, that is, methods which aim at building a graph G=(V,E), whose vertices V correspond to words in a text and edges E, connect pairs of words which co-occur in a syntactic relation, in the same paragraph, or in a larger context [4].

3 Word embedding

Recently, a lot of work has been done to represent individual words of a language as vector in a multidimensional space that conveys the semantic information contained in the words. Thanks to their ability in efficiently learning the semantics of words, these representations can serve as a fundamental unit to a wide range of Natural Language Processing. More particularly, it shows that using word vector is effective for the WSD.

In the past few years, much progress has been made on using neural networks to represent words in vector space [3] and [6].

Mikolov et al [3], proposed two new methods for building word representation in vector space using continuous bag of word (CBOW) and Skip-Gram models. These methods are based on neural network architecture.

CBOW predicts a pivot word using a window of contextual words around the pivot from the same sentence. The objective of this network architecture is to classify correctly the pivot word given to its context by using log linear classifiers [7]. On the other hand, Skip-gram models aims at training a network that predicts the likelihood of context words occurring in a given center word.

Most of the works that exploit word representations in vector space in word sense disambiguation were applied to English. However, to our knowledge, no previous work has investigated any method of representing words as vectors in Arabic word sense disambiguation.

A number of different approaches addressing the problem of word sense disambiguation based on representing words as vector in a multidimensional space have been proposed in the past few years. These are some examples:

- [8] Proposed a method to solve word sense disambiguation based on neural models. They particularly build an embedding of context by concatenating or weighting a sum of the embeddings of the words surrounding the target word. Then, sense embeddings are computed as weighted sums of the embeddings of words in the WordNet gloss for each sense.
- The method presented by [9] is a supervised learning method for word sense disambiguation based on word vector embedding of [10].The authors have shown that word embedding can be used as additional features in a supervised WSD system.

4 Proposed method

As noted earlier, in recent years, the idea of embedding words in a vector space using neural network-inspired algorithms have had significant successes in numerous NLP tasks mainly owing to their ability to capture semantic information from massive amounts of textual content. That is why word sense disambiguation has become even more prominent with the advent of neural networks as it can be efficiently solved using this method. Word embedding provided an efficient affordable method of finding similarity between different words and building semantic vector space. They require no manual annotation, only large corpora of texts, thus any set of texts can be used as a corpus.

Here, we propose to define our method for Arabic word sense disambiguation based on words embedding.

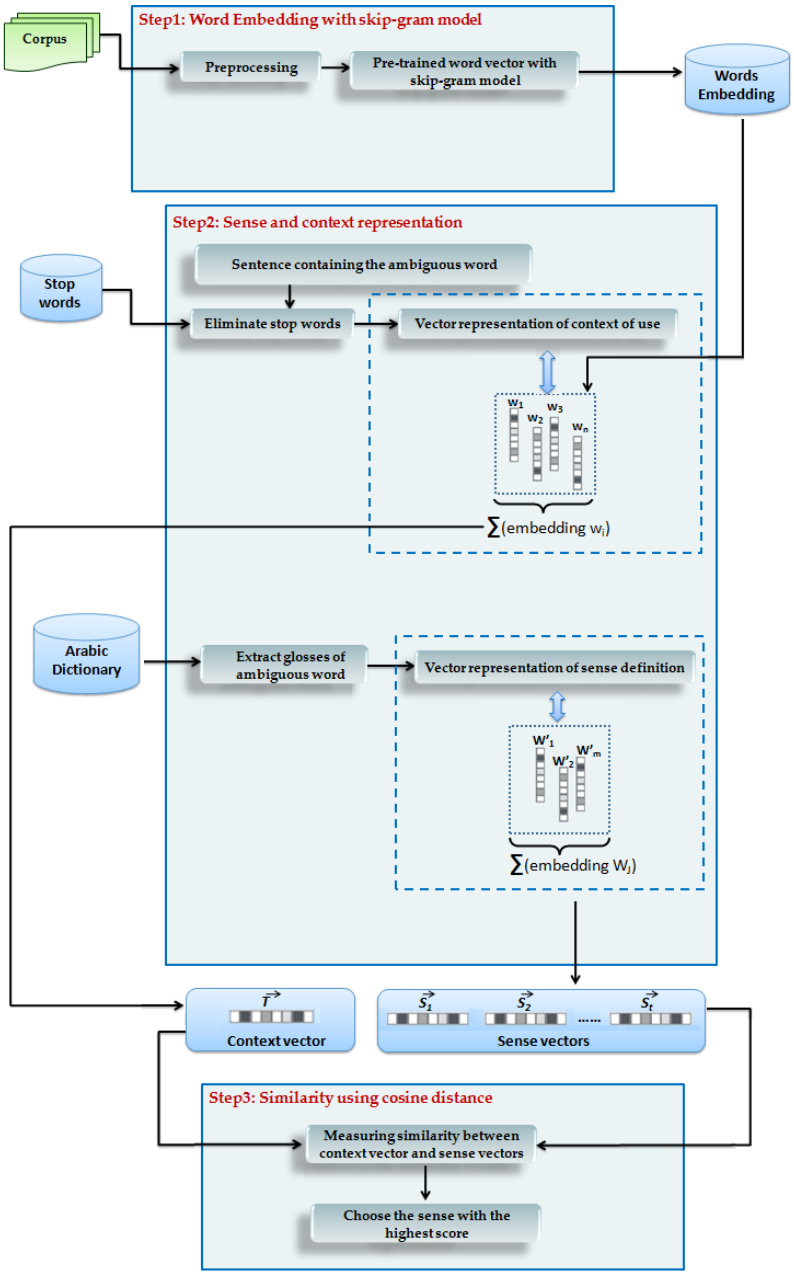
The first step of the proposed method is to train Arabic corpus. For our training corpus, we opted for Historical Arabic Dictionary Corpus [2] which is originally designed to build a historical dictionary. The dataset comprises around 86 millions words. Then, we use the Skip-gram model [3], a neural network based language model, to learn word vectors.

After learning the word vectors using Skip-Gram model, the second step of the proposed method is to assign vector representations for the context of use containing the ambiguous word and its senses based on their definitions (glosses extracted from dictionaries). Subsequently, we generate context vector and sense vectors. Our strategy of generating context vector, which is inspired by [11], consists in summing the vectors of the words surrounding a target word (we skip a word if the word is not a content word). Similar to the generating of context vector, we use the sum of all of the content word vectors in every sense definition of the ambiguous word as the generation of vectors of senses.

The last step of our proposed method is to measure the similarity between the different glosses of the ambiguous word and the current context by computing the cosine similarities between the context vector and the sense vectors of the ambiguous word. Then, we choose the sense that yields the maximum cosine similarity as an appropriate sense for the ambiguous word.

Figure 1 below describes the principle of this method.

Fig. 1. Principal of proposed method



We describe our method as a 3-steps process:

4.1 Word Embedding with Skip-gram model

We use Skip-gram to train the word vectors from large amounts of text data. We choose Skip-gram for its simplicity and effectiveness. The training objective of Skip-gram is to predict the surrounding words given the current word [3].
To train skip-gram model, we can use a large amount of raw Arabic texts from the Historical Arabic Dictionary Corpus [2]. This corpus is originally designed to build a historical dictionary, it contains texts in classical Arabic and modern standard Arabic from the 2nd up to the 21st century. There are several types of texts which can be summarized as Poetry, Quran, literary prose, Hadiths, history and genealogy, religions and doctrines, encyclopedias and dictionaries, journalistic texts, geography and travel literature.

We performed several cleaning and normalization steps to the corpus such as:

- Removing from each document in the Arabic dataset punctuation marks and diacritics.
- Normalizing the letters (آ,إ,أ) to (ا)

The vocabulary size of the compiled corpus comprises more than 86 millions of words. Training skip-gram model requires a choice of some parameters affecting the resulting vectors.

Table 1. Training configuration parameters

Parameter	Value
Vector size	300
Window	10
Sample	1e-3
Negative	10
Frequency threshold	3

- Vector size: dimensionality of the word vectors.
- Window: the amount of context to consider around the pivot word.
- Sample: threshold for sub-sampling of frequent words.
- Negative: number of negative examples in the training.
- Frequency threshold: words appearing with frequency less than this threshold will be discarded.

4.2 Sense and context representation

After learning the word vectors using the Skip-gram model, we use the content word's vectors in a sentence as the initialization vector of context. Subsequently we eliminate stop words from the original sentence, using a predefined list of stop words. In fact, stop words are eliminated as they have little semantic discrimination power in our calculation. Let $S_1 = w_{n-k},\ldots\ldots,\ldots\ldots,w_{n+k}$, be a window of text surrounding a focus word w_n, using a window size of 3 words (three words on the left and three words on the right of the ambiguous word), an embedding for the context is computed as a concatenation sum of the embeddings of the words w_i.

In order to represent the sense definition of the ambiguous word as a vector, we initialize the sense vectors based on the glosses of senses. In fact, to extract glosses of the ambiguous word we use Al-mu'jam al- wasit dictionary.

Therefore, sense vector is represented by concatenating a sum of the vectors of content words in the gloss.

4.3 Similarity using cosine distance

This last step consists in attributing for each ambiguous word its appropriate sense. This is done by choosing the sense with the closest semantic proximity to its context of use.

The degree of similarity between a sentence (containing an ambiguous word) and its sense definition is obtained by calculating cosine similarity between context vector and sense vector.

The sense definition that obtains the highest score of similarity with the current context will represent the most probable sense of the ambiguous word.

For example:

Let *W* = 'السيارة'(vehicle) be an ambiguous word and let *S* be the context of use of W:

S = قال قائل منهم لا تقتلوا يوسف والقوه في غيابة الجب يلتقطه بعض السيارة اي المارة من المسافرين ان كنتم فاعلين

(One of them said, 'Kill not Joseph, but if you must do something, cast him into the bottom of a deep well; some of the travelers will pick him up.')

We give in what follows a set of glosses for the word *W* = 'السيارة' given by the dictionary Al-mu'jam al- wasit:

First gloss:

انطلقت السيارة اي القافلة(The convoy set out)

Second gloss:

مركبة الية تسير بمحرك للبنزين تستخدم للركوب والنقل(A vehicle with an engine used for transport)

The similarity between S and each sense definition is obtained as follows:

- **Step1: context embedding.**

We use a window size of 3 words (including the ambiguous word), and we represent the context in which the word occurs as a vector by summing word vectors.

$V(S) = V(\text{الجب}) + V(\text{يلتقطه}) + V(\text{بعض}) + V(\text{السيارة}) + V(\text{المارة}) + V(\text{المسافرين}) + V(\text{فاعلين})$

- **Step2: sense embedding**.

$V_1 = V(\text{انطلقت}) + V(\text{السيارة}) + V(\text{القافلة})$
$V_2 = V(\text{مركبة}) + V(\text{الية}) + V(\text{تسير}) + V(\text{بمحرك}) + V(\text{للبنزين}) + V(\text{تستخدم}) + V(\text{للركوب}) + V(\text{والنقل})$

- **Step3: Calculate the similarity**.

$Sim(S, V_1) = cos(V(S), V_1)$
$Sim(S,V_2) = cos(V(S), V_2)$

5 Experiments and results

In order to measure effectively the performance of the proposed method, a large collection is necessary. In fact, the English works were evaluated using Senseval-1 or senseval-2. However, in our work we have to make own experimental data using a totally different set of resources.

In our experiments, we have used a test corpus containing 172 texts. From this corpus, we extracted the use contexts (examples) of each word to be disambiguated. The selection of sense for a target word was made from a list of senses given by Almu-Jam-Alwasit dictionary.

In our work, ten words were chosen. For each one of these ambiguous words we evaluated 50 examples.

We used the Word2vec toolkit[1] to learn 300 dimensional vectors. We chose the Skip-gram architecture with the negative sampling set to 10 and the window size to 10 words.

To measure the rate of disambiguation, we must use the most common evaluation techniques which select a small sample of words and compare the results of the system with a human judge. The precision measurement was used here. Experiment results have shown a precision of 78% for texts in Modern Standard Arabic.

In the table 2 bellow, we present the ten ambiguous words used in this paper and we report the statistics of the obtained precision.

Table 2. Ambiguous words used to evaluate proposed method

word	Precision
الباب	78
سيارة	82
قطار	86

[1] code.google.com/archive/p/word2vec/

عين	42
اية	78
قائم	64
دنيا	82
طائرة	96
جامع	84
حرامي	88

We can deduce from the above table that the average baseline precision is equal to 78%.

According to table 2, we can note that the weakest precision is obtained by the ambiguous word "عين"(eye). This can be explained by the fact that this word has some rarely occurring meanings which also did not frequently occur in the corpus and which are difficult to represent due to the lack of sufficient training examples.

During the disambiguation process, we have encountered the following problems:

- For some considered words, we have found out that there are some meanings which appear in the corpus but do not exist in the dictionary. For example, for the word "الباب" (door), we have extracted a dozen of sentences from the corpus where it stands for the name of a city in Syria. A sample of that is stated in the following:

  سيطر الجيش السوري الحر على مواقع حيوية بمدينة الباب أبرز معاقل تنظيم الدولة الاسلامية في ريف حلب الشرقي

  (The Free Syrian Army took control of vital sites in the city of Al-Bab, the most important stronghold of the Islamic state in Aleppo's eastern countryside).
- If a sense from the Arabic dictionary has insufficient number of words in its gloss, the vector of that sense is inaccurate.

6 Conclusion and future works

This work explores the possibility of using word embedding to solve word sense disambiguation problem. The proposed method consists of measuring semantic relation between the context of use of the ambiguous word and its senses definitions. This method carried out by training word vectors from the corpus using Mikolov's Skip-Gram model and by representing the context of a word to be disambiguated and all the senses as a vector in a multidimensional space. We apply cosine similarity to compare the similarity of the context vector with the target word sense vectors, the sense of the highest similarity being allocated as the disambiguated sense.

For a sample of 10 ambiguous Arabic words, experiments have shown a precision of 78%.

We propose that in the future works we can train our model on a larger corpus (by integrating others texts). We also propose to test our model on a larger test corpus (by adding other contexts of use for each ambiguous word) and try to integrate IDF weighing and Part-Of-Speech tagging on the context of use and senses definition in order to support the identification of words that are hightly descriptive in the examined context of use.

7 References

1. Al-Said, A. B.: Computerizing Historical Arabic Dictionary, al-Lisan al-arabi journal, al-Ribat., vol. 74 (2014)
2. Al-Said, A. B., and L. Medea--García.: The Historical Arabic Dictionary Corpus and its Suitability for a Grammaticalization Approach, 5th international conference in linguistics, Gramatyka i korpus – Grammar and Corpora, http://www.iszip.uw.edu.pl, Institute of Western and Southern Slavic Studies, University of Warsaw, Poland., (2014)
3. Mikolov, T., Sutskever, I., Chen, K., Corrado , G. S., and Dean, J.: Distributed representations of words and phrases and their compositionality. In Advances in Neural Information Processing Systems, (2013b)
4. Navigli, R.: Word sense disambiguation: a survey, ACM Computing Surveys (2009)
5. Lesk, M.: Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. The 5th annual international conference on systems documentation (1986)
6. Levy, O. and Goldberg, Y.: Neural word embedding as implicit matrix factorization. In Advances in Neural Information Processing Systems, pages 2177–2185 (2014)
7. Zahran, M., A., Magooda, A., Mahgoub, A., Y., Hazem, R., Rashwan, M., and Atyia, A.: Word representations in vector space and their applications for Arabic (2015)
8. Chen,X., Liu, Z., and Sun, M.: A unified model for word sense representation and disambiguation. In EMNLP, pages 1025–1035, (2014)
9. Taghipour, K. and Hwee Tou Ng.: Semi-Supervised Word Sense Disambiguation Using Word Embeddings in General and Specific Domains. In Proceedings of the 2015 Annual Conference of the NAACL, pages 314–323, Denver, Colorado, (2015)
10. Collobert, R. and Weston, J.: A unified architecture for natural language processing: Deep neural networks with multitask learning. In Proceedings of the 25th ICML, pages 160–167, Helsinki, Finland, (2008)
11. Nagoudi, E., M., B., and Schwab, D.: Semantic Similarity of Arabic Sentences with Word Embeddings, The Third Arabic Natural Language Processing Workshop, (2017)