International Journal of Cognitive Informatics and Natural Intelligence
Volume 14 • Issue 1 • January-March 2020

Distributional Semantic Model Based on Convolutional Neural Network for Arabic Textual Similarity

Adnen Mahmoud, Higher Institute of Computer Science and Communication Techniques, Monastir, Tunisia
Mounir Zrigui, Faculty of Science Monastir, Monastir, Tunisia

ABSTRACT

The problem addressed is to develop a model that can reliably identify whether a previously unseen document pair is paraphrased or not. Its detection in Arabic documents is a challenge because of its variability in features and the lack of publicly available corpora. Faced with these problems, the authors propose a semantic approach. At the feature extraction level, the authors use global vectors representation combining global co-occurrence counting and a contextual skip gram model. At the paraphrase identification level, the authors apply a convolutional neural network model to learn more contextual and semantic information between documents. For experiments, the authors use Open Source Arabic Corpora as a source corpus. Then the authors collect different datasets to create a vocabulary model. For the paraphrased corpus construction, the authors replace each word from the source corpus by its most similar one which has the same grammatical class applying the word2vec algorithm and the part-of-speech annotation. Experiments show that the model achieves promising results in terms of precision and recall compared to existing approaches in the literature.

KEYWORDS

Arabic Language, Context Based Approach, Global Vectors Representation, Natural Language Processing, Paraphrase Detection, Semantic Similarity, Word Embedding, Word2vec

1. INTRODUCTION

The rapid development of information and communication technologies has generated a tremendous amount of data which has increased the potential source of plagiarism. This is because of the lack of honesty, irresponsibility and self-confidence due to limited time and competitive pressure to achieve good results. It allows taking the work of others and representing it as one's own work without mentioning the source. Different ways can be applied such as directly copying ideas, adding/deleting of words, or their intelligently substituting them. In this context, we consider the problem of paraphrase detection which requires semantic textual similarity analysis. It has represented an essential problem in many Natural Language Processing (NLP) tasks (e.g. sentiment analysis, question answering, information retrieval, etc.). Often, an important problem to solve is the lack of resources in the publicly available Arabic language. The purpose of this paper is to detect Arabic paraphrase based on global Vector Representation (GloVe) as a feature extraction technique. We apply various supervised machine-learning algorithms e.g. Support Vectors Machines (SVM), Naïve Bayes (NB), Logistic Regression (LR) and Convolutional Neural Network (CNN), and we compare their performances for classification. The remainder of this paper is organized as follows: First, we present

DOI: 10.4018/IJCINI.2020010103

the problem statement in section 2. Next, we make an overview of previous work in section 3. After that, the components of our model are detailed in section 4. The experimental setup and results are discussed in section 5. Finally, we give our conclusions and future work in section 6.

2. PROBLEM STATEMENT

The amount of textual information available and stored electronically has grown at a staggering rate. This has exponentially increased the potential source of paraphrase. More formally, given two sentences S_1 and S_2, such that $S_1 \neq S_2$, when S_1 and S_2 convey the same meaning and are semantically equivalent, they are said to be paraphrased (Agarwal et al. 2017). Many researches on paraphrase detection have focused on the English language, but little effort has been done recently on other languages like Arabic. It is considered as a complex problem because of the challenging features of this language (Mohamed et al 2015). It is Semitic spoken by more than 330 million people and composed of 28 letters written from right to left. In addition, Arabic script has a rich morphologically accentuating by the existence of dots, diacritics and stacked letters (Hkiri et al. 2017, Mansouri et al. 2018, Mahmoud et al. 2018). It is highly inflectional, derivational and non-concatenative compared to other languages (Batita et al. 2018, Mahmoud et al. 2017). To contribute and solve these gaps, recent research has been advancing to propose semantic-similarity-based approaches that have more flexibility and expressiveness compared to syntactic ones. The main objective was to measure the degree of relationship between textual units and cover the maximum of Arabic specificities in terms of word construction and diversity meanings.

3. STATE OF THE ART

Word-embedding models aim a dense representation of words in the form of digital vectors and learned using a variety of language models. In addition, semantic vector representation is able to reveal many hidden relationships between words to enhance the performance of semantic computation and paraphrase detection in different languages, for instance count-based and context-based vector space models.

3.1. Count Based Vector Space Model

Count based vector space models are unsupervised. They rely heavily on the matrix of frequency and the co-occurrence of words. This is done by assuming that words in the same contexts share similar ones or related semantic meanings: Latent Semantic Analysis (LSA) based on the co-occurrence matrix makes it possible to measure the similarity between texts. It represents the meaning not only of individual words, but also of the whole passages, such as sentences, paragraphs and short texts. Based on this idea, Li et al. (2017) used Singular Value Decomposition (SVD) to reduce the dimensionality and suppress the noise of text representation models. They analyzed the optimal number of singular values and calculated the semantic relevance between words combining Term Frequency-Inverse Document Frequency (TF-IDF) weighting and cosine similarity. For experiments, Reuters-21578 data were used with 20 newsgroups and this system achieved about 0.7% of the F-measure. The Latent Dirichlet Allocation (LDA) technique has been one of the most common way of clustering texts. It was a probabilistic model for capturing polysemy (each word has multiple meanings), for example by associating a context with a document. The objective was to reduce the dimensionality of topics as it was used in the work of Dai et al. (2018). They explored semantic topics and author communities for citation recommendation. The experiments were based on the ANN and DBLP datasets and showed that this model could generate qualified author communities and topics. Furthermore, Abdelrahman et al. (2017) detected plagiarism in electronic Arabic resources using heuristic based algorithms, as follows: First, word synonyms were retrieved utilizing the WordNet dictionary. Afterwards,

fingerprints at different logical levels (document, paragraph, and sentence) were compared using the Arabic document tree. Subsequently, the Levenshtein distance and Longest Common Substring (LCS) were applied as a similarity measures.

Although these models have represented well the semantic of sentences, they have been sparse with the curse of dimensionality and loss of contextual information.

3.2. Context Based Vector Space Model

Context based methods aim to predict a given word to its neighbors where the dense word vectors are part of the model parameters. In recent decades, neural networks have achieved good results for text mining, namely distributed word vector representation (word2vec) (Mikolov et al. 2013). It consists of two different models:

- **Continuous Bag of Words (CBW) Model:** It predicts the target word from the source context words. Although this model considers the order of words in a short context and the smooth averaging step of the distributional information, it suffers from data sparsity for high dimensionality. It allows a weak semantic meaning representation of words, or more formally on the distances between them (Weng 2018);
- **Skip gram model:** While CBOW can be seen as a precognitive language model, Skip gram reverses the purpose of the language model rather than using surrounding words to predict the central word. It is an effective method for learning the high quality of distributed vector representation. It captures a large number of precise syntactic and semantic relationships for a better performance in grouping similar words (Ruder 2016).

Some work has been proposed, although there has been little work suggested for the Arabic language, to wit: Konopik et al. (2016) introduced a system for interpretable semantic textual similarity in SemEval 2016. They explored a wide variety of machine learning algorithms as well as several types of features, like: lexical (word base form overlap, word lemma overlap, chunk length difference, word sentence positions difference), syntactic (Part-Of-Speech (POS) tagging), semantic (GloVe, word2vec) and external (WordNet). The core of this system consisted in exploiting distributional semantics to compare the similarity of sentence chunks. Consequently, the combination between relation types increased the score of similarity to 0.6484 in terms of F1-measure.

Lately, several researchers have used an extensive training in sentence and classification modeling to facilitate the measurement of similarity. In the beginning, models of neural networks were useful in the literature. They were effective for NLP tasks and they achieved excellent results. Indeed, word incorporation models (e.g. word2vec, GloVe, etc.) are an in-depth learning technique that use large corpora for learning and the output contains dimensional vectors representing words. Some research studies have been developed.

Mihaylov et al. (2016) described a system for finding good answers in a community forum. They utilized the word2vec algorithm formed on different non-annotated data sources (e.g. QatarLiving and DohaNews). Subsequently, they opted for various similarity features using centroid word vectors on the body question, the subject question, and the text commentary. For experiments, they utilized an L2-regularized logistic regression classifier and obtained 69.94 precision and 73.39 accuracy. In contrast, Almarwani et al. (2017) addressed the problem of textual involvement in Arabic. They used traditional features (e.g. length of sentences and similarity scores (Jaccard and Dice), and named entities) and distributional representations (word2vec). For the immersion of words, different data were utilized, such as: Arabic Gigaword, Arabic Treebank (ATB), Arabic Wikipedia, and annotated data (ArbTE) including 600 standard modern Arabic (MSA) pairs collected from information sites and manually annotated for implication. Then, various supervised classifiers were used for prediction including SVM, LR, Random Forest. This suggested approach yielded a peak performance on the ArbTE standard dataset, reaching 76.2% accuracy. Nagoudi et al. (2018) applied the CBOW model for

relevant feature extraction. For identifying the most descriptive words, they combined it with the word alignment method, IDF and POS weighting. For learning the word2vec model, they collected multiple resources containing more than 5.8 billion words. For the evaluation task, an external Arabic plagiarism corpus was used. Multiple forms of plagiarism were developed manually, as paraphrase, substitution of synonyms, etc. This approach led to 0.8593 and 0.8781 of accuracy and recall. Similarly, Maraev et al. (2018) utilized the incorporation of characters for a morphological language like Russian. They studied the effectiveness of the CNN model for detecting a semantically equivalent issue (Convolutional filters with different lengths were concatenated). This model achieved a competitive performance for paraphrase detection without using external resources with 70.4% accuracy.

Throughout the state of the art, we have found that word vector learning models have the following limitations with respect to other models; like the global matrix factorization methods (LSA). These methods have efficiently utilized statistical information. Nevertheless, they have relatively done poorly on the word analogy spot indicating a suboptimal vector space structure. On the other hand, local pop-up methods, as the word2vec algorithm, based on Skip gram model, have done better on the analogy spot. yet, they habe badly used statistics of the corpus because it trains on distant local contextual windows instead of counting global co-occurrences. In response to these problems, we opt for the GloVe model, which will be briefly described in the following section. It combines count-based matrix factorization with contextual Skip gram models. To conduct experiments, a lack of available publicly well-structured and cleaned corpora in the Arabic language makes the evaluation and comparison between proposed solutions hard.

4. CONTRIBUTIONS FOR ARABIC PARAPHRASE DETECTION

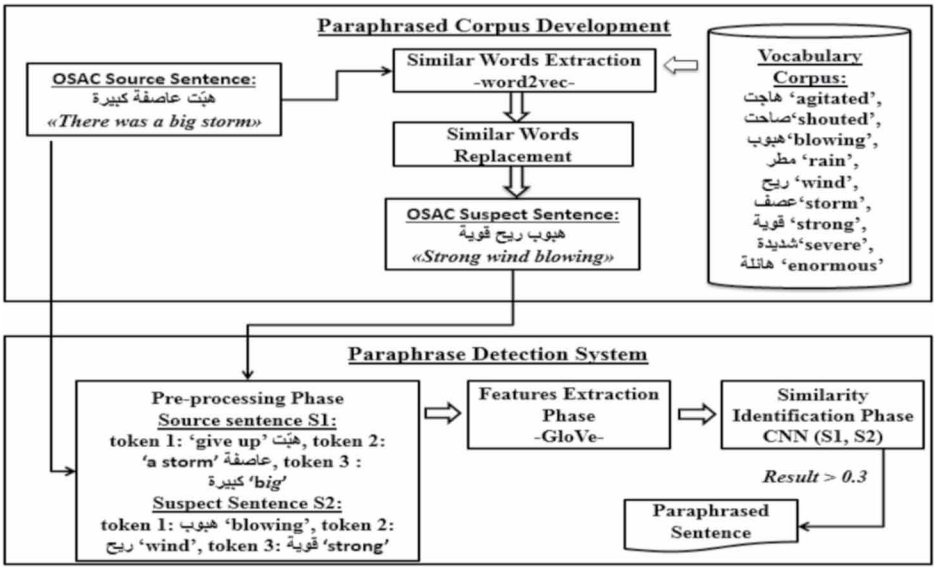
Although a wide range of shallow and deep learning techniques have matched sentence pairs, question-answer pairs, or query-document pairs, it is still challenging to model the underlying semantic similarity or relationship between Arabic documents (Liu et al. 2018). Their rich structures make it an increasingly difficult task especially when the length of documents is large. Most of previous work has focused on features, like n-gram overlapping as well as, syntactic, structural and machine translation. In recent decades, fully connected neural networks have achieved top performances for sentence modeling and classification owing to the powerful capability of capturing local relations. It learns distributed representations of words, captures more contextual information and represents the semantic of texts more precisely. The methodology adopted in this work consists in using vector representation models for feature extraction tor train different collected datasets. Various classification algorithms have been studied to develop a prediction model and detect paraphrase. Figure 1 depicts the layout of the proposed approach composed of two main components:

- Seeing the lack of resources representing different forms of obfuscations in Arabic, a paraphrased corpus is developed using local word embedding (word2vec) and syntactic annotation (POS). The objective is to conserve the syntactic and semantic properties of sentences;
- Given the complexity of Arabic paraphrase detection, the relevant and discriminative features are extracted from documents using GloVe. Subsequently, the local region information is captured in the form of important n-grams from texts, and the degree of semantic similarity is estimated via CNN model.

4.1. Arabic Monolingual Paraphrased Corpus Development

Seeing the lack of resources available publicly in Arabic, we develop an Arabic paraphrased corpus automatically, as follows: We construct paraphrased pairs expressing the same semantic content using a vocabulary corpus.

Figure 1. Proposed architecture for Arabic paraphrase detection



4.2. Vocabulary Creation

To train our model, we create a vocabulary corpus containing more than 2.3 billion words from different resources. It gathers knowledge specific about various fields in an exploitable form. Subsequently, a preprocessing step is applied describing any type of processing performed on raw data to prepare it for further processing, including:

- **Corpora cleaning:** We remove diacritics, extra white space, titles numeration, punctuation marks, special characters, duplicated letters and non-Arabic words;
- **Corpora tokenization:** We reduce the complexity of lexical and syntactic analysis, such as possessives, pronouns and discourse connectors.

For each original word w_i, we extract its synonyms from the vocabulary $\{v_1,\ldots,v_n\}$ using the distributed word vector representation algorithm (word2vec), in Equation (1):

$$word2vec(w_i) = \{word2vec(w_{v1}),\ldots,word2vec(w_{vk})\} \quad (1)$$

The Skip-Gram model is employed for learning precise relationships between words and their contexts. The goal is to improve the accuracy of capturing multiple similarity degrees along semantic dimensions.

4.3. Automatic Paraphrased Corpus Development

The second part allows representing how to randomly create the suspect corpus from OSAC composed of N words as follows.

The number of words to replace R is determined using the random uniform function that sets the degree of paraphrase U to apply in the range of 0.45 and 0.75, as illustrated in Equation (2):

$$R = N \times U \quad (2)$$

According to an index of all words in the original document between 0 and N-1, the random shuffle function replaces the index of words and keeps their content the same. To preserve both semantic and syntactic structures of Arabic sentences without ambiguities, each original word w_i is replaced by its synonym that has the same grammatical class P (e.g. verb, noun, subject, complement, etc.), as follows in Equation (3):

$$Sim\left(P\left(v_{w_i}\right),\left\{P(v_1),\ldots,P(v_k)\right\}\right) = Max\left(\left(\mathrm{Cos}\left(P\left(v_{w_i}\right),P(v_1)\right)\right),\ldots,\left(\mathrm{Cos}\left(P\left(v_{\boldsymbol{w}_i}\right),P(v_k)\right)\right)\right) \tag{3}$$

4.4. Arabic Paraphrase Detection

4.4.1. Global Vectors for Word Representation (GloVe)

The feature extraction phase selects a subset of attributes that may efficiently describe the data. Many machine-learning algorithms require that the input should be represented as a fixed-length feature vector. That is why, finding an optimal subset of features that maximizes classification accuracy is still an open problem. Therefore, we employ the GloVe model proposed by Pennington et al. (2014). The difference between this model and the best-known word2vec model is the following (Ruder, 2016):

- GloVe uses counting data while simultaneously capturing significant linear substructures prevalent in methods based on recent log-bilinear predictions, like word2vec;
- GloVe encodes meaning as vector offsets in an integration space;
- GloVe calculates the ratio of co-occurrence probabilities of two words (rather than their co-occurrence probabilities themselves) to encode their information as vector differences.

In general, we maximize the probability that a contextual word will occur, given a central word, by performing a dynamic logistic regression model. It trains in the global encoding of word-to-word co-occurrences and uses statistics efficiently to produce linear directions of meaning, as shown in Equation (4) (Vargas 2018):

$$J = \sum_{i,j=1}^{V} f\left(X_{ij}\right)\left(w_i^{\,T}\breve{w}_j + b_i + \breve{b}_j - \log\left(X_{ij}\right)\right)^2 \tag{4}$$

where X is the co-occurrence matrix, such as X_{ij} is the number of times that the word i occurs in the context of the word j; w_i and $\breve{w}_j$ are the vectors of the words i and j; b_i and $\overset{\text{ç}}{b}_j$ are the polarizations (bias) of the words i and j; V ; is the vocabulary size and f (x) is the weighting function that assigns a relatively lower weight to rare and frequent co-occurrences.

4.4.2. Convolutional Neural Network (CNN)

A CNN is feed-forward architecture characterized by local connections, shared weights among different locations and local pooling. It can be more suitable to employ word embedding as an input with their particular architecture designs. The main idea is to consider the contextual relationship between words and encode all interactions in a general parameter (Wang et al. 2018). It is a successful model for extracting high-level abstract features from sentences of different n-grams (Gua et al. 2017). In our work, the representations of source and suspect corpora are utilized as entries in the CNN model. They are processed in parallel, as depicted in Figure 2 through the following layers: A convolution

layer extracts the useful features from Arabic documents. A max-pooling layer reduces the number of connections between convolution layers. A fully connected layer computes the rate of paraphrase and converts the output score into probability.

4.4.3. Sentence Modeling Layer

Given a sequence of words $w_{1:n} = w_1,\ldots,w_n$, each one is represented with an embedding vector of dimension k. Considering a window of words $w_i, w_{i+1}\ldots, w_{i+s}$, we employ a convolution layer to produce a feature map $S\left[i\right]$ by sliding 64 filters over the input for each $w_s = \left\{2,3,4\right\}$. At every region, a matrix multiplication W is performed with an addition of a bias term b, which is then followed by a nonlinear activation function h, as shown in Equation (5):

$$S\left[i\right] = h\left(W \cdot x_{i:i+w_s-1} + b\right) \tag{5}$$

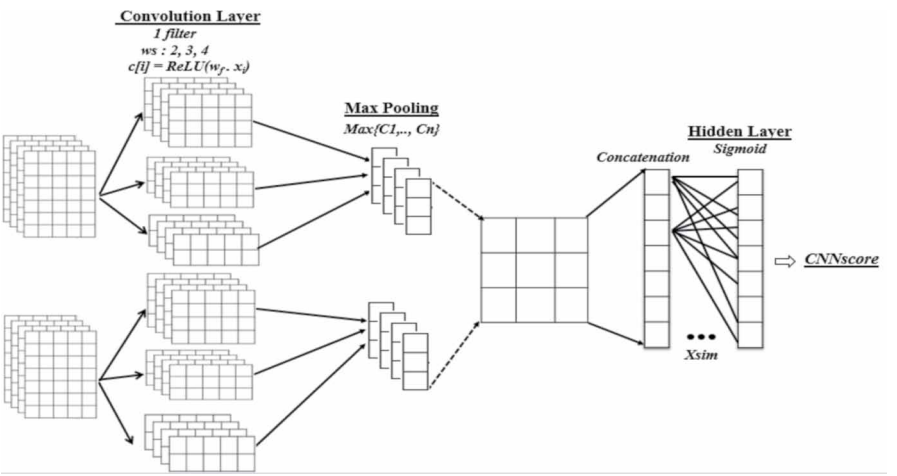
where $W \in R_{w_s * k}$ is the weight vector of the filter reducing the model complexity and making the network easier to train and h is the activation function of words $\left\{x_{1:w_s}, x_{2:w_s+1},\ldots, x_{n-w_s+1:n}\right\}$. In our model, we use the Rectified Linear Unit (ReLU) to induce sparsity in hidden units and obtain sparse representations, defined in Equation (6):

$$R_{i,j,k} = \max\left(z_{i,j,k}, 0\right) \tag{6}$$

4.4.4. Reduction of Features Layer

The pooling layer sweeps a rectangular window over the input vectors. It aims to achieve shift-invariances by reducing the dimensionality of the feature maps and the number of parameters (Wang et al. 2018). Each feature map of the pooling layer is connected to its corresponding one of the preceding convolution layers. We take the maximum value in the resulting vectors to capture the most relevant

Figure 2. Proposed model for semantic textual similarity identification



feature. Our goal is to reduce the representation complexity and assume the maximum value as a feature corresponding to this filter, as represented in Equation (7):

$$P_l = Max\left\{S\left[i\right]\right\} \tag{7}$$

where i: 1, …, n-j+1 denotes the number of convolutions, and l is the number of sentences. All results are concatenated to form a single feature vector for the penultimate layer, in Equation (8):

$$Pooling_{Vector} = P_1,\ldots,P_n \tag{8}$$

As a result, we obtain two reduced feature vectors that correspond to source and suspect sentences for detecting paraphrase in the following layer.

4.4.5. Similarity Layer

After several convolution and pooling layers, our classification model performs a high level of reasoning and generates global semantic information. It takes all neurons in the previous layer and connects them to all the activation functions of the current layer. We propose a binary classification model applying the sigmoid function to convert the output score into probability in Equation (9). When the obtained degree is higher than threshold α, a pair of sentences are considered paraphrased (class 1). Otherwise, they are considered unparaphrased (class 0):

$$Output = Sigmoid\left(x\right) = \sigma\left(x\right) = \frac{e^x}{\left(1+e^x\right)} \tag{9}$$

5. EXPERIMENTS AND DISCUSSION

5.1. Data Used

Experiments are carried out on two different datasets, as summarized in Table 1:

- To develop the vocabulary model, we collect corpora from different resources, such as Arabic Corpora Resource (AraCorpus), King Saud University Corpus of Classical Arabic (KSUCCA) (Alrabiah et al. 2014) and papers from Wikipedia;
- To develop the paraphrased model, we use OSAC (Saad et al. 2010) as a source corpus. It includes 22,429 documents of different fields, such as Economics, History, Entertainment, Education & Family, Religion and Fatwas, Sports, Health, Astronomy, Law, Stories, and Cooking Recipes.

Indeed, the evaluation of our model is carried out on a collection of documents randomly used from the OSAC source corpus: 70% are original for training and 30% are paraphrased for testing.

5.2. Used Parameters

5.2.1. Building Word-Embedding Models

The word2ec algorithm based on Skip gram and GloVe models are efficient for capturing semantic relations between words after studying different configurations.

Table 1. Dataset summary

Models	Corpora	Number of Words
Vocabulary model	AraCorpus	126,026,301
	KSUCCA	48,743,953
	Wikipedia	2,158,904,163
	Total Number	2.3 billion
Test model	OSAC	18,183,511

5.2.2. Word2vec Parameters

For developing the paraphrased corpus, we vary the parameters of the word2vec model to conserve the semantic structure of Arabic sentences as provided in Table 2.

An average of all cosine similarities of words in each sentence is calculated to test the effectiveness of our proposed method, as given in Equation (10). Regarding the constraint that we have proposed above, the resulted paraphrased sentences preserve the syntactic structure of original sentences with semantically similar words, as illustrated in the example in Table 3:

$$Sen2vec_i = \frac{\sum_{i=1}^{n} w_i}{n} \tag{10}$$

Finally, we make some modifications to suit the model with Arabic encoding and the dataset for training and testing. Figure 3 depicts different configurations of word2vec according to vector dimensions and widow sizes to efficiently develop a paraphrased sentence.

The experimental settings of the word2vec algorithm, based on the Skip gram model, are investigated with various configurations. The best parameters for identifying the semantically closest words of the target word are the following: the vector dimension is 300, the slippery window size is 3, and the maximum number of iterations is 7, as shown in Table 4.

5.3. GloVe Parameters

At the level of Arabic paraphrase detection, we conduct our experiments with several settings for the parameters of the GloVe model. The following parameters make the results efficient for capturing global semantic features as presented in Table 5.

5.4. CNN Parameters

Our CNN model consists of the following components:

- Different widow sizes are studied to obtain multiple features. When the sliding window size increases, a longer n-gram is extracted from the input sentences. The best results are obtained

Table 2. Training parameters of word2vec model

Vector Dimensions	Window Sizes
64, 100, 250, 300, 350, 400, 500	1, 2, 3, 4, 5, 6, 7, 8, 9, 10

Table 3. Example of paraphrased sentence development

Original words	word2vec (P(Synonym), Cos)						
اكتشاف	ابتكار	ابتكار	اختراع	اختراع	خلق	اصدار	انتاج
'discovery'	'innovation'	'innovation'	'invention'	'invention'	'create'	'issuing'	'produce'
'noun'	'noun'	'noun'	'noun'	'noun'	'noun'	'noun'	'noun'
	0.78	0.78	0.85	0.85	0.8	0.7	0.75
جديد	حديث	حديث	حديث	حديث	حديث	حديث	جدي
'new'	'modern'	'modern'	'modern'	'modern'	'modern'	'modern'	'serious'
'adjective'	'adjective'	'adjective'	'adjective'	'adjective'	'adjective'	'adjective'	'adjective'
	0.87	0.87	0.87	0.87	0.87	0.87	0.65
يعين	يساعد	يجدد	يساعد	يساعد	يعيد	يعيل	يشير
'identifies'	'helps'	'renews'	'helps'	'helps'	'restores'	'depends'	'indicates'
'verb'	'verb'	'verb'	'verb'	'verb'	'verb'	'verb'	'verb'
	0.88	0.65	0.88	0.88	0.60	0.79	0.64
ال+معاقين	إعاقة	ال+قصر	ال+مرضى	ال+معاق	ال+مرضى	ال+محتاج	مرض
'd*isabled*	'Disability'	'problem'	'patients'	'the	'patients'	'the	'illness'
people'	'noun'	'noun'	'noun'	disabled'	'noun'	needy'	'noun'
'noun'	0.7	0.76	0.75	'noun'	0.75	'noun'	0.83
				0.83		0.72	
ال+تنقل	ال+عبور	ال+تجول	الإنتقال	ال+تحرك	ال+مشي	الإنتقال	ال+مشي
'mobility'	'transit'	'getting'	'moving'	'moving'	'walking'	'moving'	'walking'
'noun'	'noun'	'noun'	'noun'	'noun'	'noun'	'noun'	'noun'
	0.66	0.82	0.8	0.85	0.7	0.8	0.7
Sen2vec	0.778	0.776	0.83	0.856	0.744	0.776	0.714
Suspect sentence	اختراع حديث يساعد ال+معاق على ال+تحرك						
	'Modern invention helps the disabled on moving'						

Figure 3. Configurations of word2vec model

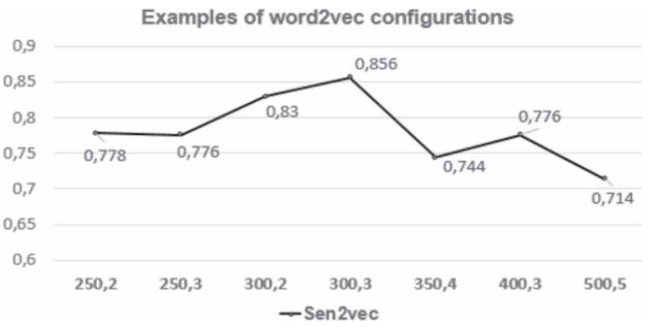


Table 4. Parameters of word2vec model

Parameters	word2vec
Vector Size	300
Min_Count	≤ 5
Window Size	3
Workers	8
Epochs	7

Table 5. Training parameters of GloVe algorithm

Parameters	Values
Size of co-occurrence Matrix	1.119.436 *1.119.436 words
Embedding size	300
Context size	3
Minimum occurrence	25
Learning rate	0.05
Batch size	512
Numbers of Epochs	20

when we combine multiple window sizes $\left(w_s = \{2,3,4\}\right)$ in the convolution layer, where 64 filters are moved for each region, as depicted in Figure 4:

- A pooling layer of size 4 calculates the maximum pooling of each sentence;
- Two sentences are considered as paraphrase if they exceed threshold (α). The threshold is fine-tuned by several trials on the training corpus, and the results are achieved when α=0.3.

The best parameters of the proposed CNN model are provided in Table 6.

5.5. Results and Discussion

5.5.1. Performance Evaluation

The evaluation measures are defined as follows:

- Precision represents the number of correct instances over the number of correctly predicted instances as in Equation (11):

Figure 4. Configurations of widow size

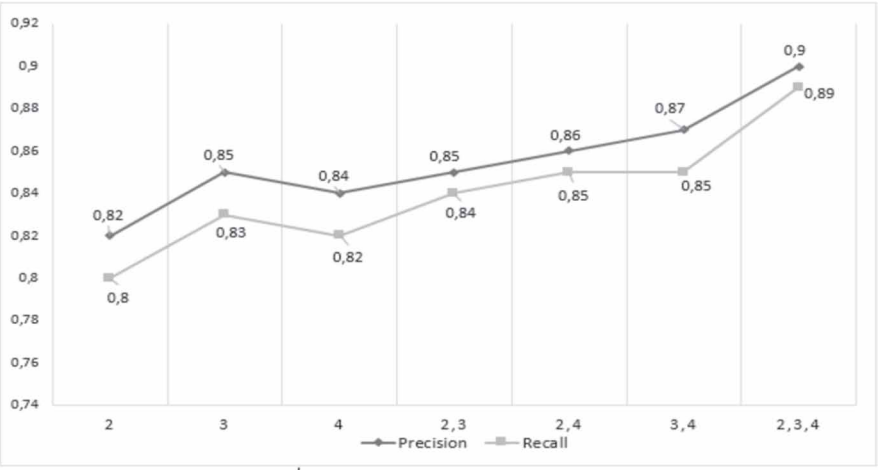


Table 6. Parameters of CNN model

CNN Layers	Parameters	Values
Convolution layer	Number of filters	64
	Kernel size	2, 3, 4
	Activation function	ReLU
Pooling layer	Type	Max pooling
	Pooling size	4
Fully connected layer	Activation function	Sigmoid

$$Precision = \frac{\#\ of\ correct\ instances}{\#\ of\ correctly\ predicted\ instances} \tag{11}$$

- Recall represents the number of correct instances over the number of true instances as in Equation (12):

$$Recall = \frac{\#\ of\ correct\ instances}{\#\ of\ true\ instances} \tag{12}$$

The experiments show that the performance of any paraphrase detection system depends on the quality of analyzed data (morphology, syntactic and semantic structures, etc.) and the adopted methodology. Table 7 illustrate how our proposed methods outperform the state-of-the-art methods in terms of precision (82%) and recall (80%).

Indeed, an Arabic paraphrased corpus is developed automatically using local word embedding and POS techniques. It conserves the syntactic structures of original sentences and modifies them semantically with similar ones. Consequently, different obfuscation forms such as total copying, synonym substitution, word/sentence shuffling are created. Using this dataset, GloVe is advantageous in capturing significant linear substructures than recent log-bilinear prediction methods like word2vec and other models (e.g. LSA and LDA) for analogy reasoning and semantic similarity analysis. These models are weak for representing semantic relations between words when data are rarely present or their number goes up. For training and testing, we study the performance of statistic classifiers that are widely used in machine learning and data mining compared to the CNN model. SVM, LR and NB are utilized with default parameters. Indeed, the detection rate of paraphrase rises by 79% using SVM compared to NB (80%) and LR (76%). For a very large number of learning data, the calculation time explodes. Thus, SVM is useful for small classification problems. In contrast, the CNN model efficiently supportes this problem. It is useful to deal with long sentences through different regularities of window sizes, learning more contextual information and represent hidden semantic relations between documents.

Seeing the lack of work developed for Arabic paraphrase detection, our system is compared to other existing models in the literature based on various datasets and working on the same topic of semantic textual similarity analysis in the Arabic language: The system proposed by Nagoudi et al. (2018) was efficient in capturing the most relevant features in documents. They combined the CBOW model, the inverse document frequency weighting and the POS methods for extracting semantic and syntactic features from documents. Subsequently a word alignment method was applied for measuring

similarity. However, this system was weak in representing data with higher dimensionality. It was limited for working on distant local contextual windows instead of counting global co-occurrences, as proposed in our work. To compute similarity and ranking candidate Arabic Question-Answer QA pairs, Abdel-Latif et al. (2018) combined lexical (e.g. term/sentence overlap) and semantic features (e.g. weighted matrix factorization and Fasttext) between original question and each QA pair. They also used three types of learning models, such as the SVM rank, the classification algorithms (e.g. linear SVM, LR, Random Forest and stochastic gradient descent) and the deep learning approach (fully-connected deep neural network)). The best learning algorithm was the fully connected neural network. However, this did not hold for the test set because the training dataset was not large enough for the deep neural network to have a better generalization effect.

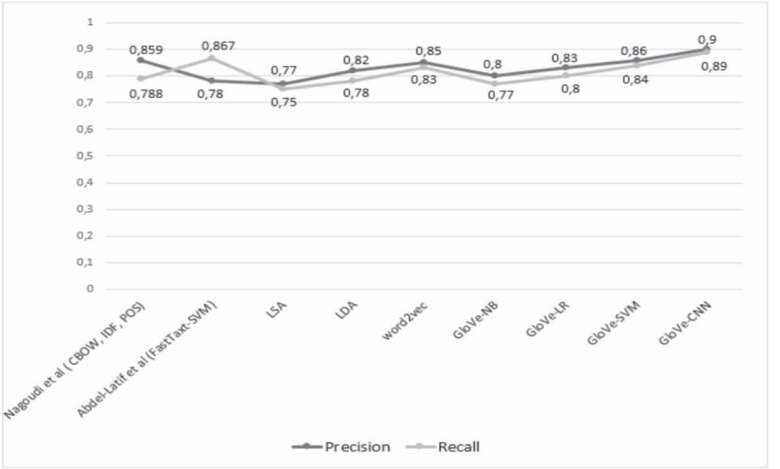
6. CONCLUSION

The use of the word2vec algorithm and the POS weighting method has resulted in a well-structured paraphrased corpus. Then the combination of GloVe and CNN has outperformed the state-of-the-art methods and has efficiently realized semantic analysis for external paraphrase detection in Arabic documents. Indeed, GloVe algorithm has been efficient in capturing contextual information from documents. Thereafter, the CNN model has successfully identified the paraphrase between source and suspect documents using the advantages of their convolution, pooling and fully-connected layers. Despite the promising results, we will try to use other deep-learning-based models to improve paraphrase detection, especially in Arabic. The objective is to capture more statistical regularities in the context of sentences, like the long short-term memory recurrent neural network.

Table 7. Comparative study

References	Models	Datasets	Precision	Recall
Proposed models	LSA	OSAC source OSAC suspect	0.773	0.765
	LDA		0.782	0.780
	Word2vec, CNN		0.810	0.804
	GloVe, NB		0.793	0,770
	GloVe, LR		0.763	0.740
	GloVe, SVM		0.803	0.784
	GloVe, CNN		0.820	0.805
Nagoudi et al. (2018)	CBOW, IDF, POS, word alignment	External Arabic Corpus	0.859	0.788
Abdel-Latif et al. (2018)	Lexical and semantic features, SVM	Arabic CQA dataset	0.78	0.867

Figure 5. Overall comparisons in terms of precision and recall



REFERENCES

Abdel-Latif, M., Samir, M., Abdel-Aziz, M., Heeba, M., Elmasry, A., & Torki, M. (2018). A supervised learning approach using the combination of semantic and lexical features for Arabic community question answering. *15th ACS/IEEE International Conference on Computer Systems and Applications (AICCSA)*, 1-7. doi:10.1109/AICCSA.2018.8612828

Abdelrahman, Y. A., Khalid, A., & Osman, I. M. (2017). A method for Arabic documents plagiarism detection. *International Journal of Computer Science and Information Security*, *15*(2), 79–85.

Agarwal, B., Ramampiaroa, H., Langsetha, H., & Ruoccoa, M. (2017). *A deep network model for paraphrase detection in short text messages*. arXiv:1712.02820 [cs. IR]

Almarwani, N., & Diab, M. (2017). Arabic textual entailment with word embeddings. *The Third Arabic Natural Language Processing Workshop (WANLP)*, 185–190. doi:10.18653/v1/W17-1322

Alrabiah, M., Al-Salman, A., Atwell, E., & Alhelewh, N. (2014). KSUCCA: A key to exploring Arabic historical linguistics. *International Journal of Computational Linguistics*, *5*(2), 27–36.

Batita, M. A., & Zrigui, M. (2018). Derivational relations in Arabic Wordnet. *The 9th Global WordNet Conference (GWC)*, 137-144.

Dai, T., Zhu, L., Cai, X., Pan, S., & Yuan, S. (2018). Explore semantic topics and author communities for citation recommendation in bipartite bibliographic network. *Journal of Ambient Intelligence and Humanized Computing*, *9*(9), 957–975. doi:10.1007/s12652-017-0497-1

Gua, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., . . . Wang, G. (2017). *Recent advances in Convolutional Neural Networks*. arXiv:1512.07108 [cs. CV]

Hkiri, E., Mallat, S., & Zrigui, M. (2017). Arabic-English text translation leveraging hybrid NER. *The 31nd Pacific Asia Conference on Language, Information and Computation (PACLIC 31)*, 124-131.

Konopik, M., Prazak, O., Steinberger, D., & Brychcín, T. (2016). UWB at SemEval-2016 Task 2: Interpretable semantic textual similarity with distributional semantics for chunks. *10th International Workshop on Semantic Evaluation (SemEval-2016)*, 803–808. doi:10.18653/v1/S16-1124

Li, X., Yao, C., Fan, F., & Yu, X. (2017). A text similarity measurement method based on singular value decomposition and semantic relevance. *Journal of Information Processing Systems*, *13*(4), 863–875.

Liu, B., Zhang, T., Niu, D., Lin, J., Lai, K., & Xu, Y. (2018). *Matching long text documents via graph Convolutional Network*. arXiv:1802.07459 [cv.CL]

Mahmoud, A., Zrigui, A., & Zrigui, M. (2017). A text semantic similarity approach for Arabic paraphrase detection. *18th International Conference on Computational Linguistics and Intelligent Text Processing (CICLing)*, 338-349. Doi:10.1007/978-3-319-77116-8_25

Mahmoud, A., & Zrigui, M. (2017). Semantic similarity analysis for paraphrase identification in Arabic texts. *The 31st Pacific Asia Conference on Language, Information and Computation, Philippine, (PACLIC 31)*, 274-281.

Mahmoud, A., & Zrigui, M. (2018). Artificial method for building monolingual plagiarized Arabic corpus. *Computación y Sistemas*, *22*(3), 767–776. doi:10.13053/cys-22-3-3019

Mansouri, S., Charhad, M. M., & Zrigui, M. (2018). A heuristic approach to detect and localize text in Arabic news video. *Computación y Sistemas*, *23*(1), 75–82.

Maraev, V., Saedi, C., Rodrigues, J., Branco, A., & Silva, J. (2018) Character-level Convolutional Neural Network for paraphrase detection and other Experiments. *Conference on Artificial Intelligence and Natural Language Journal*, 1-13. doi:10.1007/978-3-319-71746-3_23

Mihaylov, T., & Nakov, P. (2016). SemanticZ at SemEval-2016 Task 3: Ranking relevant answers in community question answering using semantic similarity based on fine-tuned word embeddings. SemEval-2016, 879–886.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. *26th International Conference on Neural Information Processing Systems*, 2, 3111–3119.

Mohamed, M. A. B., Mallat, S., Nahdi, M. A., & Zrigui, M. (2015). Exploring the potential of schemes in building NLP tools for Arabic language. *The International Arab Journal of Information Technology*, *6*(12), 13–19.

Nagoudi, E. B., Khorsi, A., Cherroun, H., & Schwab, D. (2018). A Two-level plagiarism detection system for Arabic documents. *Cybernetics and Information Technologies*, *18*(1), 124–138. doi:10.2478/cait-2018-0011

Ruder, S. (2016). *An overview of word embeddings and their connection to distributional semantic models*. Retrieved from http://blog.aylien.com/overview-word-embeddings-history-word2vec-cbow-glove/

Saad, M. K., & Ashour, W. (2010). OSAC: Open Source Arabic Corpora. *Proceedings of the 6th International Conference on Electrical and Computer Systems EECS'10*, 1-6. Doi:10.13140/2.1.4664.9288

Vargas, E. (2018). *A Comprehensive Introduction to Word Vector Representations*. Retrieved from https://medium.com/ai-society/jkljlj-7d6e699895c4

Wang, W., Zhou, M., & Fei, G. (2018). *Contextual and position-aware factorization machines for sentiment classification*. arXiv:1801.06172 [cs.CL]

Weng, L. (2018). *Learning word embedding*. Retrieved from https://lilianweng.github.io/lil-log/2017/10/15/learning-word-embedding.html#glove-global-vectors

Zhang, L., Wang, S., & Liu, B. (2018). Deep learning for sentiment analysis: A survey. *Data Mining and Knowledge Discovery*, *8*(4). doi:10.1002/widm.1253

Zrigui, S., Zouaghi, A., Ayadi, R., Zrigui, M., & Zrigui, S. (2016). ISAO: An intelligent system of opinion analysis. *Research in Computing*, *110*, 21–31.