Arabian Journal for Science and Engineering
https://doi.org/10.1007/s13369-019-04039-7

RESEARCH ARTICLE - COMPUTER ENGINEERING AND COMPUTER SCIENCE

Sentence Embedding and Convolutional Neural Network for Semantic Textual Similarity Detection in Arabic Language

Adnen Mahmoud[1,2] · **Mounir Zrigui**[1]

Received: 14 December 2018 / Accepted: 16 July 2019

Abstract
The continuous increase in extraordinary textual sources on the web has facilitated the act of paraphrase. Its detection has become a challenge in different natural language processing applications (e.g., plagiarism detection, information retrieval and extraction, question answering, etc.). Different from western languages like English, few works have been addressed the problem of extrinsic paraphrase detection in Arabic language. In this context, we proposed a deep learning-based approach to indicate how original and suspect documents expressed the same meaning. Indeed, word2vec algorithm extracted the relevant features by predicting each word to its neighbors. Subsequently, averaging the obtained vectors was efficient for generating sentence vectors representations. Then, convolutional neural network was useful to capture more contextual information and compute the degree of semantic relatedness. Faced to the lack of resources publicly available, paraphrased corpus was developed using skip gram model. It had better performance in replacing an original word by its most similar one that had the same grammatical class from a vocabulary. Finally, the proposed system achieved good results enhancing an efficient contextual relationship detection between Arabic documents in terms of precision (85%) and recall (86.8%) than previous studies.

Keywords Arabic language · Paraphrase detection · Semantic similarity analysis · Sentence vector representation · Convolutional neural network · Natural language processing

1 Introduction

The heavy use of electronic technologies has increased the act of paraphrase by expressing original and suspect documents with the same meanings without citing the source [1, 2]. This could be applied with different structures, semantics or contextual representations including rewording, synonym substitution, text manipulation, text translation or idea adoption [3, 4].

In recent decades, paraphrase detection has become a challenge seeing the widespread of textual reuse. It has emphasized in various contexts and has impeded the pertinence of automatic natural language processing [5]. Therefore, machine learning algorithms have attracted the interest of scientific community in different text mining applications (e.g., plagiarism detection, question answering classification, information extraction and retrieval, etc.) [6]. Thus, several textual similarity detection systems have been developed. We distinguish those that have been focused on multilingual data by translating a text from a different language and then have been integrated it into their own work. Others have been concentrated on the specificities of data in the same language.

Nevertheless, Arabic paraphrase detection is a fundamental issue in natural language processing (NLP). This language is rich of morph-syntactic and semantic features that complicate its analysis. In this context, we propose a context-based approach for detecting monolingual paraphrases in Arabic language using distributed semantic vector spaces. Our objective is to extract the most relevant features

✉ Adnen Mahmoud
mahmoud.adnen@gmail.com

Mounir Zrigui
mounir.zrigui@fsm.rnu.tn

[1] Algebra, Numbers Theory and Nonlinear Analyzes Laboratory LATNAL, University of Monastir, Monastir, Tunisia

[2] Higher Institute of Computer Science and Communication Techniques, Hammam Sousse, University of Sousse, Sousse, Tunisia

Published online: 26 July 2019

by analyzing the context of words to determine thereafter the meaning of the whole sentence. Then, we estimate the correspondence between paraphrased documents and a reference collection comprising a set of source documents.

The main contributions of this paper are the following:

- Automatic development of a paraphrased corpus preserving the properties of Arabic language. This is by combining skip gram and part-of-speech techniques.
- Feed-forward neural network architecture for sentence modeling and semantic textual similarity computation.

The outline for the rest of this paper is the following: Sect. 2 provides an overview of the background. Section 3 presents a state of the art in the field of paraphrase identification and details the complexities of Arabic language. Section 4 describes the proposed methodology. Section 5 gives a brief description of the experimental setup including data preparation, evaluation and discussion. Section 6 presents the conclusion and the suggestions for the future work.

2 Background

Humans can understand the differences between contextual information, which is an extremely difficult task. In contrast, automatic text mining and artificial intelligence work more efficiently with a wide amount of structured data [7].

2.1 Distributed Word Vector Representation

Neural networks have been regained popularity to train models using distributed word vector representation (word2vec) proposed by Mikolov et al. [8]. It is a necessary step to carry semantic meanings and group similar words in a continuous space of predefined size. Indeed, word2vec algorithm introduces two different models [9]:

- Continuous bag of words (CBOW) is a bigram model predicting one target word given its context.
- Skip gram model is efficient to learn and capture a large number of precise semantic and syntactic relationships. It predicts the surrounding words of a target one in a sentence or a document.

To judge paraphrases, source and suspect documents should be mapped in feature vectors with a fixed length. The objective is to identify thereafter the semantic similarity between them. However, paraphrase phenomenon needs an appropriate semantic representation in matching natural language sentences, which is a vital problem for the following reason: the same idea can be expressed with words in different orders or contexts.

2.2 Convolutional Neural Network

Deep learning models have achieved remarkable results in computer vision and speech recognition in recent years. Later, they have involved the learning of word vector representations through neural language models and have performed their composition for classification [8, 10]. Currently, these models have achieved excellent results and outperformed traditional NLP models (e.g., latent semantic analysis, latent Dirichlet allocation (LDA), term frequency-inverse document frequency (TF-IDF), etc.) in semantic parsing, query retrieval, sentence modeling, etc.

The most useful deep learning architecture is convolutional neural networks (CNN) for sentence modeling and semantic analysis. It has the ability to learn different data structures automatically through the following layers:

- *Convolutional layer* is the core block of CNN to produce new invariant features. It can extract different matching patterns using multiple filter widths and feature maps applied to a window of words
- *Pooling layer* is applied over the resulted feature maps at the convolutional layer. It captures the most important features and reduces the computational cost of their representations. Different pooling methods can be applied, such as minimum, maximum and average.
- *Fully connected layer* combines all the feature maps from the previous layers. It is capable to generate the output of the CNN network by learning the complex nonlinear interactions.

3 Overview on Paraphrase Identification

The similarity analysis is an important issue to make its detection more intelligent to predict. Such new approaches can rely on concepts from the areas of NLP and text mining [11]. In this section, we present the challenges related for paraphrase detection in Arabic documents. Then, we cite several existing works in literature for this task.

3.1 Problematic

The rapid development of technology has improved the quantity and the complexity of information and has made difficult to choose the relevant ones that meet the needs of the user. In addition, this fact has enabled not only reusing texts but also stealing concepts and ideas without mention the original sources [3, 11]. Unlike the English language, Arabic language is Semitic. It is the fifth most used language in the world and the mother tongue of over 200

million peoples [12]. Arabic is among the languages that characterized by complex morphological aspects and lack both linguistic and semantic resources [13].

Because words can have more than one meaning and ideas can be stated in multiple ways, paraphrase detection has considered as a very difficult task. Therefore, it has needed a specific process for computing the score of relatedness between source and suspect documents including the information extraction and semantic similarity methods. This challenge increased with Arabic language. It was difficult to treat automatically because of its great variability of morphological, syntactic and semantic specificities that explain its sparseness [14]:

- *Morphologically complex language:* the existence of dots, diacritics and stacked letters above or below the baseline of words [15, 16].
- *Inflected complex language:* lexical units vary in number and in bending according to the grammatical relationships [17].
- *Agglutinative complex language:* a word may have several possible divisions (proclitic, flexive and enclitic forms), which increases the ambiguity of word segmentation, including low-quality assessment, and time complexity [18, 19].
- *Non-concatenative complex language:* the morphology of words corresponds to the modification of the internal structure of a word in different grammatical categories (e.g., noun, verb, adjective, etc.) [20]. Therefore, a word can have different meanings of words in the sentence [21, 22].
- *Derivational complex language:* the morphology of Arabic words corresponds to nouns, active/passive particles and other derivations based on pattern changes [23, 24].

The identification of Arabic paraphrases represents a serious challenge because of the amount of data publicly available. Therefore, developing an efficient system is increasingly a necessity for this language.

3.2 Related Works

Paraphrase detection based on relatedness measurement is a general concept that includes semantic similarity. It consists of combining semantic comparisons of sentence kernel elements (subject, verb and object) in order to estimate the overall similarity [25]. Following the literature, semantic similarity between documents could be divided into lexical, syntactic and semantic measures [26]:

- Lexical and syntactic measures take into consideration the sequencing and the order of words/characters in comparing the linguistic units (words, sentences, paragraphs, documents).
- Semantic measures overcome the limitations of the syntactic ones by comparing linguistic units according to their semantics. To compare the linguistics units, the semantic measures in general could be classified as corpus or knowledge-based approaches. The corpus-based approach uses unstructured semantic data; while the knowledge-based approach uses the structured semantic data like ontologies.

Several works have been proposed for paraphrase detection in different languages based on textual similarity analysis, distinguish:

Al-Shenak et al. [27] enhanced a method for Arabic question answering. They used latent semantic analysis (LSA) for modeling terms and documents to the same concept space and support vector machines (SVM) for classification. To authenticate the answers precisely, Shehab et al. [28] proposed an automatic Arabic essay grading system with a comparative study of different similarity algorithms like string (Damerau–Levenshtein and *N*-gram) and corpus (LSA, latent Dirichlet allocation (LDA) and DISCO). For experiments, 210 students' Arabic answers were used. This system achieved a best result with *N*-gram algorithm of 0.82 correlation. In the same idea, Imran et al. [29] developed a framework for extrinsic plagiarism avoidance in research articles using *N*-gram features (between three and five) and Dice coefficient for similarity computation. In addition, Rafiq et al. [30] detected plagiarism in Urdu documents applying the following NLP techniques: tokenization, stop word removal, chunking (trigram) and hashing (absolute hashing). Furthermore, Aburaig et al. [31] collected Arabic papers and annotated them with different versions of political orientations. Experiments denoted the superiority of the traditional text categorization over the stylometric features-based approaches. The highest accuracies obtained by combining partition membership (PM) feature selection method and SVM classifier.

In contrast, other systems have been focused on knowledge-based approaches in their researches. They have been quantified the semantic similarity and relatedness in the context of concepts, paragraphs, terms and documents:

The case of abstract meaning representation (AMR) parsing is a problem, in which one is required to specify whether two sentences have the same meaning. Therefore, Issa et al. [32] combined LSA technique and AMR parsing. This method significantly advanced the state-of-the-art results in paraphrase detection for Microsoft Research Paraphrase Corpus (MRPC). Results achieved 86.6% accuracy and 90.0% F1 measure. However, El-Deeb et al. [26] focused on semantic relatedness in short texts. They proposed a vector space model based on multi-corpus. Thus,

word synonyms and anaphoric information improved the semantic representation of the document. Then, they used a set of verses in the Holy Quran as the main case study to measure the degree of relatedness between them. Experiments showed an improvement to the recall to be 60% rather than 11.3% as the previous studies. In contrast, Ezzikouri et al. [33] proposed a fuzzy-semantic similarity for cross-language plagiarism detection using WordNet taxonomy and three semantic approaches such as Wu and Palmer, Lin and Leacock–Chodorow for Arabic documents. In the same idea, Fernando et al. [34] presented an algorithm for paraphrase identification using word similarity information derived from WordNet. For experiments, they used MRPC. This approach achieved 75.2% precision and 91.3% recall. Moreover, Mihalcea et al. [35] computed the semantic similarity between short texts, using corpus-based and knowledge-based measures of similarity. Experiments showed that the semantic similarity method outperformed methods based on lexical matching, resulting in up to 13% error rate reduction with respect to the traditional vector-based similarity metric.

Traditional NLP approaches have been denoted continuous representations of words (e.g., bag of words (BoW), term frequency-inverse document frequency (TF-IDF), latent semantic analysis (LSA), latent Dirichlet allocation (LDA), etc.). They have not taken the syntactic structure of language into consideration. That is why they could not capture the similarity between words [36]. Therefore, competitive works have been learned on distributed representations of words by using neural networks models, in recent years. They have been useful to preserve the linear regularities among words, alleviate data sparseness and learn large datasets.

To detect semantic difference of concepts pairs, Lai et al. [37] proposed a SVM-based method combined features extracted from pretrained global embedding (GloVe) and statistical information from Is-A taxonomy. Otherwise, Nagoudi et al. [3] devoted various approaches to detect plagiarism in Arabic texts. The first approach employed word2vec algorithm based on skip gram model, words alignment and weighting to measure the semantic similarity relationships. The second approach studied machine learning methods at the sentence level. Then, they combined lexical, syntactic and semantic features to assist the detection task. The training and evaluation based on SVM, decision trees (DT) and random forest (RF) classifiers using the first Arabic plagiarism detection (AraPlagDet) shared task 2015.

On the other hand, Kim [10] described a series of experiments with CNN model built on the top of word2vec embedding for sentence classification with little hyper-parameter tuning (i.e., static and non-static channels, multiple filter widths and feature maps). Consequently, static vectors achieved excellent results on multiple benchmarks in sentiment analysis and question–answering systems. Similarly, He et al. [38] identified English semantic textual similarity in SemEval-2016 competition. They developed an attention-based input interaction layer and incorporated it into multi-perspective CNN using the paragram-phrase word embedding trained on paraphrased pairs. Without using any sparse features, this model evaluated on the STS2015 data and achieved 80.1% Pearson. In addition, Salem et al. [39] clustered segments of Arabic texts and found which had a different stylometry in comparison to the other using CNN model as a classifier.

Throughout the state of the art, we found a lack of works to detect paraphrase in Arabic language despite that it represents a challenge in the fields of text mining and semantic information retrieval. It is coming back to its richness of specificities in terms of word's construction and diversity meanings [40].

In this context, we focus on analyzing the properties of Arabic sentences to increase the performance of paraphrase detection in the following sections.

4 Proposed Methodology

Paraphrasing original sentences allows rewriting their contents with different words in the same meaning. Often, neural networks outperformed traditional methods for text classification. In this section, we describe briefly the proposed methodology composed by the following components: First, word2vec algorithm encodes documents into vectors to represent the relevant features. Then, a feed-forward architecture based on convolutional neural network (CNN) model learns high-level features adaptively and computes thereafter the semantic similarity.

Figure 1 shows the proposed architecture, and the main processes are described below.

4.1 Document Representation

Continuous bag of words (CBOW) and count-based approaches cannot represent the semantic of language and risk of data sparsity problem. Therefore, we consolidate the representation of documents using skip gram model. It is advantageous in predicting the context of words and training systems on large corpora quickly. Its objective is based on the unique representation of words in a surrounding window as input and tries to predict the context of the middle word as represented in Eq. (1):

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c, j\neq 0}\log p(w_{t+j}|w_t) \tag{1}$$

where T and c correspond to the number of words in the vocabulary and the context size that can be a function of the central word w_t; and $p\left(w_{t+j}|w_t\right)$ is the Softmax function defined as follows in Eq. (2):

$$p\left(w_O|w_I\right)=\frac{\exp\left(v'_{w_O}\mathrm{T}v_{w_I}\right)}{\sum_{w=1}^{W}\exp\left(v'_{w}\mathrm{T}v_{w_I}\right)} \tag{2}$$

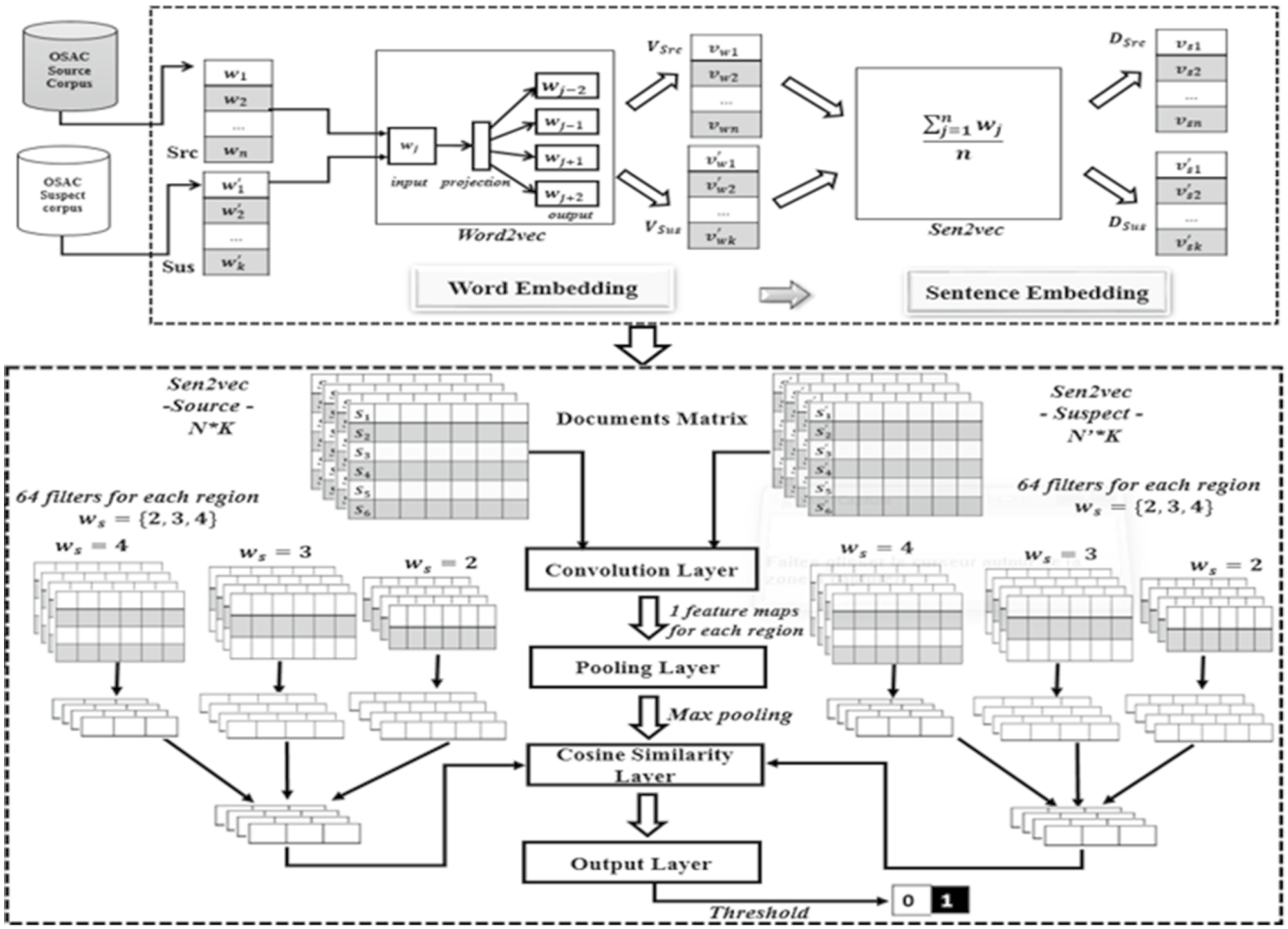

Fig. 1 Proposed architecture for contextual relatedness detection in Arabic documents

where the input v_w and the output v'_w are the vector representations of the target word w and W is the number of words in the vocabulary.

The resulted vectors of a given sentence are mapped into a matrix M of size $N \times K$, as follows in Eq. (3):

$$M_S = v_{w_1}, v_{w_2}, \ldots, v_{w_n} \tag{3}$$

Until now, we predict the context of words given the current word vector. To improve the learning quality of sentences, we extent this by averaging all its words vectors from M_S, as shown in Eq. (4):

$$Sen2vec(S_j) = \frac{\sum_{i=1}^{n} v_{w_i}}{n} \tag{4}$$

As a result, we obtain a feature maps of H sentences of a given document in a matrix D of a fixed size $H * K$, as follows in Eq. (5):

$$D_{1:H} = Sen2vec(S_1), \ldots, Sen2vec(S_h) \tag{5}$$

The algorithm 1 describes the process of vector representation of each sentence as shown in Fig. 2.

4.2 Context-Based Approach for Arabic Paraphrase Identification

Convolutional neural network (CNN) model is applied to learn (Sen2vec) sentences embeddings of source and suspect documents as inputs and extract thereafter high level of abstract features from different n-grams. It is a feed-forward architecture characterized by local connections, shared weights among different locations and local pooling. Our proposed model consists of analyzing the contextual relationship and encodes all semantic interactions, as follows: Indeed, a convolution layer extracts the useful features from documents and transfers them to a more proper form by CNN. Then, a max-pooling layer generates a reduced semantic vector. Thereafter, a comparator layer evaluates the similarity between sentences vectors and converts the output score into a probability distribution. Figure 3 details briefly the process of Arabic paraphrase identification:

Algorithm1: Distributed Sentence Vector Representation

Input: Sentence S_i of N words, Vocabulary size T, window size c.

Output: Matrix M'_D of H sentences vectors

Begin

For each sentence S_i do

1. According to T and c, predict the surrounding words of a target one w_i in S_i applying Skip gram algorithm:
$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\le j\le c, j\neq 0}\log p(w_{t+j}|w_t)$$
2. Map word vectors v_{w_i} in a matrix M_s of size $N \times K$:
$$M_S = v_{w_1}, v_{w_2}, \ldots, v_{w_n}$$
3. Average all words vectors of S_i:
$$Sen2vec(S_j) = \frac{\sum_{i=1}^{n} v_{w_i}}{n}$$
4. Map sentences vectors v'_{S_i} of a document in a matrix M'_D of size $H \times K$:
$$M'_D = v'_{s_1}, v'_{s_2}, \ldots, v'_{s_h}$$

End

Fig. 2 Distributed sentence vector representation algorithm

Algorithm2: Arabic Paraphrase Identification

Input: Sentences vectors representations $M'_{D_i} = v_{s_1}, v_{s_2}, \ldots, v_{s_n}$ and $M'_{D_j} = v'_{s_1}, v'_{s_2}, \ldots, v'_{s_n}$ for source and suspect documents.

Output: Binary classification

Begin

For each sentence vector x_i in a given matrix M'_D do

1. Extract the most invariant features applying convoluted filters and different window sizes w_s through a convolutional layer:
$$h_i = \mathrm{ReLU}(Wx_{i:i+w_s-1} + b)$$
2. Store the new feature maps:
$$C_i = [S_1, \ldots, S_{n-j+1}], C \in R^{n-j+1}$$
3. Identify the most descriptive features $P = [p_1, p_2 \ldots, p_n]$ applying max-pooling layer:
$$p_i = \max_{1\le i\le n-w_s+1} C_i$$
4. Measure the degree of semantic similarity using cosine:
$$\mathrm{Cos}(A, B) = \frac{A.B}{\|A\|\|B\|}$$
5. Average all obtain values to determine the global score S of similarity in the range of [0, 1].
6. According to a threshold α, identify the existence of paraphrase :

If $S >$ α, pair of document are paraphrased:

Output = 1

Otherwise, pair of document are not paraphrased

Output = 0

End

Fig. 3 Arabic paraphrase identification algorithm

4.2.1 Sentence Modeling Layer

We extract the most useful information using multiple configurations of convolutional filters and window sizes. The rectified linear unit (ReLU) is used as an activation function in the convolutional layer in order to obtain sparse representation. Given different window sizes $w_s = \{2, 3, 4\}$, the input documents matrix are processed by a convolutional layer h_i with 64 filters for each region to produce feature maps C_i. A weight vector $W \in R_{j*k}$ is shared to decrease the complexity of training and a bias term for polarization $b \in R$. The output of this layer is defined as follows in Eqs. (6) and (7):

$$h_i = ReLU(WD_{i:i+j-1} + b) \tag{6}$$

$$C_i = [S_1, \ldots, S_{n-j+1}], C \in R^{n-j+1} \tag{7}$$

The most descriptive features are extracted by applying the max-pooling layer as illustrated in Eq. (8). It produces a reduced feature maps $P = [p_1, p_2 \ldots, p_n]$ to simplify the complexity of further processing, where:

$$p_i = \max_{1\le i\le n-w_s+1} C_i \tag{8}$$

4.2.2 Semantic Textual Similarity Measurement Layer

The computation of semantic similarity between each pair of documents D_1 and D_2 depends on the cosine of the angle between their corresponding pair of vectors (A, B). It may be better due to the difference in the length of documents, as represented in Eq. (9) [32]:

$$\mathrm{Cos}(A, B) = \frac{A.B}{AB} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} * \sqrt{\sum_{i=1}^{n} (B_i)^2}} \tag{9}$$

The obtained results are averaged to determine the global score of similarity. It is in the range of [0, 1]. In the case that this degree is higher than a threshold α, pair of document are considered paraphrased.

Table 1 Arabic corpora collected

Datasets	Words number
Vocabulary model	
KSUCCA [41]	48,743,953
AraCorpus[a]	126,026,301
Wikipedia[b]	2,158,904,163
Total number	2.3 billion
Test model	
OSAC[c]	18,183,511

[a]http://aracorpus.e3rab.com/

[b]https://fr.wikipedia.org/wiki/Wikip%C3%A9dia_en_arabe

[c]http://site.iugaza.edu.ps/msaad/osac-open-source-arabic-corpora/

5 Experimental Setup

5.1 Arabic Paraphrased Corpus Building

The lack of public and structured resources has represented a challenge for evaluating Arabic paraphrase detection systems. Therefore, we develop our own corpus in an automatic way by proceeding as follows:

Different datasets are collected to form the knowledge database and source corpus as shown in Table 1.

The documents are preprocessed as follows:

- Remove irrelevant data such as extra white spaces, titles numeration and non-Arabic words.
- Normalize words such as "أ،إ،آ،ٱ" to "ا" and "ة" to "ه" to reduce ambiguities.
- Divide each document into tokens by detecting the space between words for simplifying their exploration.
- Annotate words with their grammatical classes (e.g., verb (V), noun (N), adjective (ADJ), etc.) using the Stanford Parser tool.[1] It is efficient to capture the syntactic structure of Arabic language and facilitate further processing.

The paraphrased dataset should contain diverse obfuscations forms that are lexically similar but not convey the same meaning (semantically dissimilar). Following the literature, word2vec algorithm based on skip gram model has gained competitive results in analogy reasoning. It is efficient for representing semantic similarity/relations between words to exploit. Therefore, we use it for extracting the synonyms of each original word from the vocabulary model.

The most likely sentences to be paraphrases are selected randomly through the following process:

1. Make the degree of paraphrase D in the range of $[0.45, \ldots, 0.75]$ using random uniform[2] function. Respecting it, source and candidate sentences are paraphrased with more than four words. When it is less than 45%, they are similar (copy and paste). Otherwise ($D > 75\%$), they are different.
2. Using this rate D, calculate the number of words to replace N in the source corpus of size K as defined in Eq. (10):

$$N = D \times K \tag{10}$$

3. Replace the index of words and keep their content the same using random shuffle[3] function. To preserve the semantic and syntactic properties, paraphrased sentence must have the same grammatical structure with synonym words compared to the original one (Table 2).

The evaluation of our proposed approach is carried out on the Open Source Arabic Corpora (OSAC) dataset as a source corpus in which 30% from its content is paraphrased randomly. Table 3 shows the statistics of the corpus:

5.2 Parameters Configurations

Table 4 details the configurations of word2vec and CNN models that performed our approaches:

5.3 Performances Measures

The evaluation measures are defined as follows [42]:

Precision is the number of correct instances over the number of correctly predicted instances:

$$P = \frac{\text{Number of correct instances}}{\text{Number of correctly predicted instrances}} \tag{11}$$

Recall is the number of correct instances over the number of true instances:

$$R = \frac{\text{Number of correct instances}}{\text{Number of true instrances}} \tag{12}$$

F1 score is the harmonic mean of precision and recall values that brings the balance between them, as follows in Eq. (13):

$$F1 = \frac{2 * P * R}{P + R} \tag{13}$$

[1] https://nlp.stanford.edu/software/.

[2] https://docs.scipy.org/doc/numpy/reference/generated/numpy.random.uniform.html.

[3] https://docs.scipy.org/doc/numpy-1.13.0/reference/generated/numpy.random.shuffle.html.

Table 2 Degree of paraphrase configuration

Source	Arabic	نرى أن الاقتصاد العالمي يمر في الوقت الراهن بمرحلة انتقالية	
	English	We see that the world economy is currently in transition	
N		Example	
$< N$ Copy and paste		نرى أن الاقتصاد العالمي يمر في الوقت الراهن بمرحلة انتقالية	We see that the world economy is currently in transition
$= N$ More than four words are paraphrased		نلاحظ أن الاقتصاد الدولي يمر في الأوان الحالي بطور انتقالي	We note that the international economy is currently undergoing a transition phase
$> N$ Dissimilarity		نلاحظ أن العالم يشهد في الأوان الحالي وضعا اقتصاديا غير ثابت	We note that at present the world is witnessing an unstable economic situation

Table 3 Statistics of the corpus

	#Documents	#Paraphrase	#Different
Train	15701	4710	10991
Test	6728	2019	4709

Table 4 Configurations of word2vec and GloVe models

Models	Parameters	Values
Word2vec	Vector dimension	300
	Window size	3
	Vocabulary size	2.3 billion
	Workers	8
	Epochs	7
	Min_count	25
CNN	Window sizes	2, 3, 4
	Activation function	ReLU
	Filters number	64
	Pooling function	Max
	Pooling size	4
	Threshold	0.3

5.4 Results and Evaluation

5.4.1 Paraphrased Corpus Analysis

Table 5 illustrates an example representing the process of paraphrased sentence construction. Indeed, we note that our proposed combination conserved the structure of Arabic language as shown in the sentence $Suspect_2$, while we consider an ambiguity problem of tense in $Suspect_1$ when we applied only word2vec. For example, the original word "العالمي" is an adjective (ADJ). It is replaced by a noun in the resulted sentence "عالم." In this way, we increased the quality and precision of our resulted corpus.

Table 5 Example of paraphrased sentence construction

Original	Arabic	نرى أن الاقتصاد العالمي يمر في الوقت الراهن بمرحلة انتقالية		
	English	We see that the world economy is currently in transition		
Tokens			Synonyms	
Arabic	English	TAG	Arabic, Cos	English
نرى	see	V	نلاحظ(0.83)|ورى(0.84)|ادراك(0.78)	Note |sight |catch sight of
أن	that	PREP	أن	that
الاقتصاد	economy	N	اقتصاد	economy
العالمي	world	ADJ	الكلي(0.77)|الدولي(0.85)|عالم(0.91)	Micro |international |world
يمر	pass	V	|(0.84) يخضع عبر(0.79)| مر 0.86))	undergoing |go over | go along
في	in	PREP	في	in
الوقت	the time	N	اوان(0.85)|احيانا(0.75)|غالبا(0.77)	Time |perhaps |often
الراهن	current	ADJ	الحاضر(0.82) |(0.82) الحالي | 0.76)(اللازم	Present |currently |necessary
ب	by	PREP	ب (0.98)	by
مرحلة	phase	N	|(0.80)وضع |(0.78)طور 0.72)رحل	Situation |transition phase |gone
انتقالية	transition	ADJ	غير ثابت |(0.83)غير مستقر (0.83)	Precarious |uncertain
$Suspect_1$	Arabic	ورى أن اقتصاد عالم مر في الأوان الحاضر بوضع غير ثابت		
	English	He saw that the economy of a world at present was in a precarious situation		
$Suspect_2$	Arabic	نلاحظ أن الاقتصاد الدولي يخضع في الأوان الحالي بطور انتقالي		
	English	We note that the international economy is currently undergoing a transition phase		

Different configurations of skip gram model are studied including window sizes and vector dimensions. Thus, we measure the semantic similarity between the resulted documents using cosine similarity. Figure 4 shows the test accuracy curves with various window sizes and vector dimensions. The x-axis is the window sizes, and the y-axis is the cosine ratio. It is clear that the parameters 3 as window size and 300 as vector dimension are the most appropriate for constructing efficiently the suspect corpus with 0.86 cosine score.

For analyzing the quality of our resulted corpus, we propose a binary paraphrase judgment in the range of [0, 1] in terms of precision and recall as shown in Table 6. We note that our corpus represents efficiently the structure of Arabic language. It contains paraphrases that are lexically and semantically diverse.

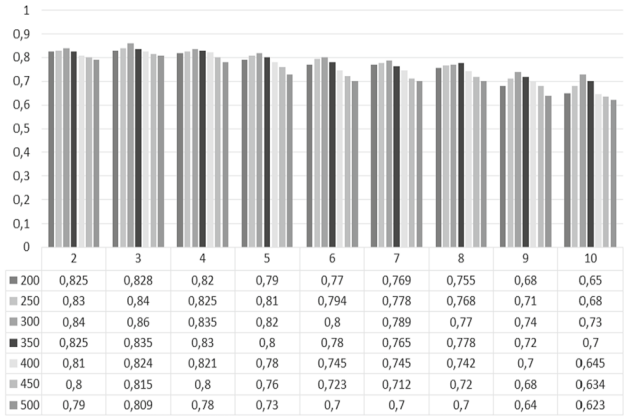

Fig. 4 Paraphrased corpus construction regarding cosine

Table 6 Binary judgment of the resulted corpus quality

Topics	Precision	Recall
Economics	0.780	0.761
History	0.850	0.830
Education and Family	0.869	0.838
Religious and Fatwas	0.819	0.800
Sports	0.825	0.798
Health	0.792	0.780
Astronomy	0.882	0.810
Low	0.872	0.842
Stories	0.798	0.781
Cooking recipes	0.843	0.830
Overall	0.833	0.807

5.4.2 Discussion

Throughout our experiments, we concluded the following observations:

Although word2vec algorithm was useful for capturing the context of words, it was worse with small corpus due to the use of negative sampling that depended on the corpus size. Furthermore, the number of learning epochs specified a single-pass program through the data. This made a change for several passes a trivial task. Nevertheless, Tables 7 and 8 demonstrate the effectiveness of word2vec model. It was capable to work on distant local contextual windows instead of counting global co-occurrences. It was useful to capture the semantic of Arabic sentences with a window size of 3 and vector dimension of 300. This configuration achieved the highest scores of 83.2% precision and 84.2% recall.

Compared to word2vec model, we investigate the performance of Sen2vec and CNN models for sentence modeling and similarity computation:

Table 7 Performance regarding precision

Vector dimensions	Window sizes			
	2	3	4	5
250	0.755	0.772	0.740	0.712
300	0.785	0.832	0.775	0.732
350	0.772	0.780	0.772	0.725
400	0.745	0.759	0.732	0.682

Table 8 Performance regarding recall

Vector dimensions	Window sizes			
	2	3	4	5
250	0.735	0.752	0.725	0.698
300	0.763	0.842	0.745	0.716
350	0.750	0.752	0.732	0.700
400	0.721	0.738	0.697	0.659

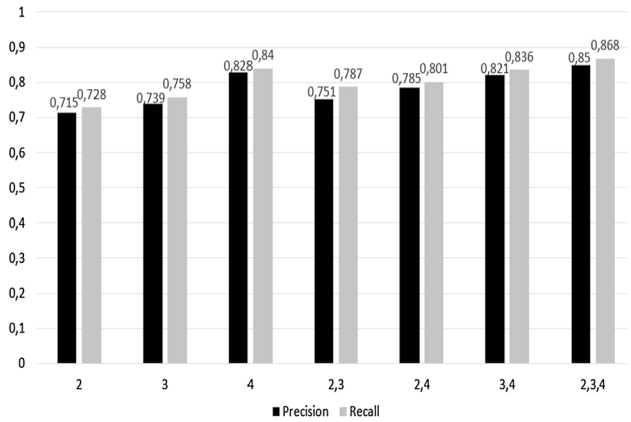

Fig. 5 Performance of Sen2vec-CNN model regarding window sizes study

Sen2vec method was able to bridge lexical gaps and information limit by the use of the average of all word vectors representations. It distinguished the meaning semantically of a large dataset with a fixed dimensionality of vectors on the embedded space than traditional methods like LDA, LSA, etc.

Moreover, the use of CNN model found specific combination patterns via convolutional operations with different window sizes around a neighborhood of inputs (vectors of sentences). To cover the maximum of Arabic sentences structures with different lengths, we used three window sizes $w_s = \{2, 3, 4\}$ with 64 filters for each. This amplified the influence of high-quality features to measure efficiently the resemblance between documents. After 50 training iterations, Fig. 5 shows the test curves with different window sizes. The x-axis is the window sizes, and the y-axis is the

Table 9 Comparative evaluation

Models	Datasets	Description	*P* %	*R* %	*F* %
Our models	(OSAC) Source and Paraphrased corpora	LDA, Cosine	80.7	82	84.3
		LSA, Cosine	75	77	76.2
		Word2vec, CNN	83.2	84.2	83.6
		Sen2vec, CNN	85	86.8	85.8
Mohsen et al. [44]	14148 Arabic documents	TF-IDF, Cosine	53	72	61
Sameen et al. [43]	2684 short Urdu text pairs	*N*-gram overlap, RJ48	–	–	70.4 ($n=3$) 77.5 ($n=5, 6$)
Almarwani and Diab [45]	ArbtE	Word2vec, SVM, LR, RF	–	–	76.2

precision and recall of the CNN models on the test corpus. It is clear that $w_s = 4$ is the most appropriate window size, which gives 82.8% precision and 84% recall for training CNN model with individual window. The overall experimental results denote the good benefit of their combination $w_s = \{2, 3, 4\}$ for capturing more contextual features within sentences. It gives the maximum precision of 85% and recall of 86.8%.

Table 9 and Fig. 6 demonstrate that our system outperformed the state-of-the-art methods in terms of precision, recall and F1 score.

Sameen et al. [43] proposed an Urdu short text reuse corpus. It contained 2684 short Urdu text pairs, manually labeled as verbatim (496), paraphrased (1329) and independently written (859). Thereafter, they detailed an evaluation of their corpus using various reuse detection methods, including lexical methods (word *n*-gram overlap and vector space model), string and sequence alignment methods (longest common subsequence (LCS), greedy string tiling (GST), global and local alignments), structural methods (character *n*-gram overlap) and stylistic methods (token ratio and type token ratio). Experiments showed that character *n*-gram overlap outperformed was efficient for Urdu short text reuse detection. However, paraphrase aims that the texts must be semantically the same but rephrased using, but not limited to, addition/deletion of words, synonym substitutions, lexical changes, active to passive switching, etc. That is why it is necessary to propose a system that should be capable to capture semantics for better textual similarity identification.

By providing semantic relationships between terms allowing to expand or alter user queries, thesauri can help in retrieving items that are more relevant. In this context, Mohsen et al. [44] collected 14,148 Arabic documents on different topics such as arts and politics. The dataset analyzed to assign weights to each term using three approaches: TF-IDF, pointwise mutual information (PMI) and LSA. Then, three different similarity measures (Cosine, Jaccard and Dice) computed similarity. Then, they tested the constructed thesauri on 20 queries to evaluate their accuracies and determine which combination performed the best.

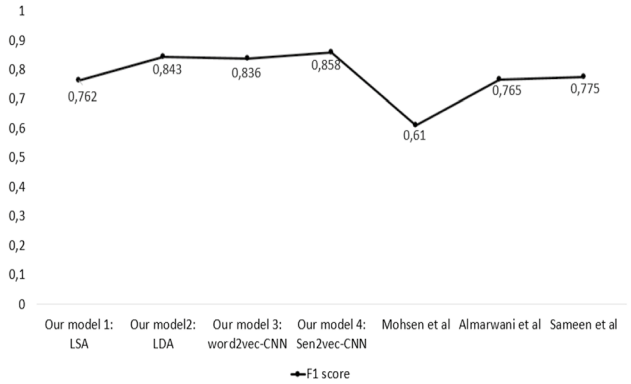

Fig. 6 Evaluation comparison with other systems regarding F1 score

Experimental results demonstrated the superiority of TF-IDF and Cosine similarity over PMI and LSA methods.

Compared to other existing systems based on distributed word vectors representations, we mention:

Almarwani and Diab [45] used both traditional features (length of sentences and similarity scores (Jaccard and Dice) and named entities) and distributional representation like word2vec. Thus, the main feature was the fact that they did not depend on external linguistic resources, but induced in the latent space, using word2vec template. It could be easily generated in any language an advantage over the use of external resources. For experiments, they used different datasets such as Arabic Gigaword, Arabic Treebank (ATB) and Arabic Wikipedia, and annotated data (ArbTE) (including 600 standard modern Arabic pairs collected from information sites) and manually annotated for implication. Subsequently, they used different types of supervised classifiers such as SVM, logistic regression (LR) and random forest (RF).

These models did better on the analogy spot. However, they used the statistics of the corpus badly because they trained on distant local contextual windows instead of counting the global co-occurrences. Therefore, the use of CNN model captured more efficiently local relations with fewer parameters, especially for document modeling.

To sum up, our proposed system based on sentence vector representation (Sen2vec) and window-based features through CNN model achieved promising results in terms of precision (85%) and recall (86.8%). Consequently, the quality of the data used and the consistence of the methodology adopted were two factors to increase the performance of any paraphrase detection system.

6 Conclusion

Paraphrase detection aims a semantic similarity analysis to determine the degree of correspondence between documents. However, Arabic language represents a challenge in this task because of the great complexity and richness of its specificities.

To address this issue, we proposed a context-based approach for monolingual Arabic paraphrase detection. We reduced the computational complexity and the data sparsity problem using word2vec algorithm. The obtained vectors averaged thereafter to generate a sentence vector representation (Sen2vec). Then, we applied CNN model with different statistic regularities for document modeling and semantic similarity measurement. To conduct our experiments, we developed an Arabic paraphrased corpus based on word-2vec algorithm seeing the lack of publicly available Arabic paraphrased resources. We replaced each word from the OSAC source corpus by its most similar one that had the same grammatical class from the vocabulary. Experiments showed promising results of 85% precision and 86.8% recall.

Although CNN model worked better for extracting invariant features, its performance influenced by the window size, in which large window size led to data sparsity and changed the semantic of sentence. Therefore, we try to integrate recurrent neural networks architectures in future works. They do not take into consideration this constraint and represent long sentences dependencies in less time. Our main goal is to improve the performance of our system trying more statistical regularities in the context of sentences and documents.

References

1. Yang, S.; Guo, J.; Wei, R.: Semantic interoperability with heterogeneous information systems on the internet through automatic tabular document exchange. Inf. Syst. **69**, 195–217 (2017). https://doi.org/10.1016/j.is.2016.10.010
2. Choi, D.; Chung, C.: A K-partitioning algorithm for clustering large-scale spatio-textual data. Inf. Syst. **64**, 1–11 (2017). https://doi.org/10.1016/j.is.2016.08.003
3. Nagoudi, E.M.B.; Cherroun, H.; Alshehri, A.: Disguised plagiarism detection in Arabic text documents. In: 2nd International Conference on Natural Language and Speech Processing (ICNLSP), pp. 1–8 (2018). https://doi.org/10.1109/icnlsp.2018.8374395
4. Xu, W.; Callison-Burch, C.; Dolan, W.B.: SemEval-2015 Task 1: paraphrase and semantic similarity in twitter (PIT). In: 9th International Workshop on Semantic Evaluation (SemEval 2015), Denver, Colorado, pp. 1–11 (2015)
5. Van der Aa, H.; Leopold, H.; Reijers, H.A.: Checking process compliance against natural language specifications using behavioral spaces. Inf. Syst. **78**, 83–95 (2018). https://doi.org/10.1016/j.is.2018.01.007
6. Salles, T.; Rocha, L.; Mourao, F.; Goncalves, M.; Viegas, F.; Meira Jr., W.: A two-stage machine learning approach for temporally-robust text classification. Inf. Syst. **69**, 40–58 (2017). https://doi.org/10.1016/j.is.2017.04.004
7. Kumar, V.; Verma, A.; Mittal, N.; Gromov, S.V.: Anatomy of pre-processing of big data for monolingual corpora paraphrase extraction: source language sentence selection. In: Emerging Technologies in Data Mining and Information Security, pp. 495–505. Springer, Singapore (2018)
8. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.S.; Dean, J.: Distributed representations of words and phrases and their compositionality. In: 26th International Conference on Neural Information Processing Systems, vol. 2, pp. 3111–3119 (2013)
9. Rong, X.: Word2vec parameter learning explained. arXiv :1411.2738 [cs] (2014)
10. Kim, Y.: Convolutional neural networks for sentence classification. In: Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1746–1751 (2014). https://doi.org/10.3115/v1/d14-1181
11. Al-Sabahi, K.; Zhang, Z.; Long, J.; Alwesabi, K.: An enhanced latent semantic analysis approach for Arabic document summarization. Arab. J. Sci. Eng. **43**, 8079–8094 (2018)
12. AlZu'bi, S.; Hawashin, B.; ElBes, M.; Al-Ayyoub, M.: A novel recommender system based on apriori algorithm for requirements engineering. In: Fifth International Conference on Social Networks Analysis, Management and Security (SNAMS), pp. 323–327. IEEE (2018)
13. Mahmoud, A.; Zrigui, M.: Artificial method for building monolingual plagiarized Arabic corpus. Computacion y Sistemas **22**, 3767–3776 (2018)
14. Moawad, I.; Alromima, W.; Rania, E.: Bi-gram term collocations-based query expansion approach for improving Arabic information retrieval. Arab. J. Sci. Eng. **43**, 7705–7718 (2018)
15. Zrigui, S.; Zouaghi, A.; Ayadi, R.; Zrigui, M.; Zrigui, S.: ISAO: an intelligent system of opinion analysis. Res. Comput. **110**, 21–31 (2016)
16. Mahmoud, A.; Zrigui, M.: Semantic similarity analysis for paraphrase identification in Arabic texts. In: The 31st Pacific Asia Conference on Language, Information and Computation, Philippine, (PACLIC 31), pp. 274–281 (2017)
17. Hkiri, E.; Mallat, S.; Zrigui, M.: Arabic–English text translation leveraging hybrid NER. The 31st Pacific Asia Conference on Language, Information and Computation (PACLIC 31), pp. 124–131 (2017)
18. Mansouri, S.; Charhad, M.M.; Zrigui, M.: A heuristic approach to detect and localize text in Arabic news video. Computacion Sistemas **23**(1), 75–82 (2018). https://doi.org/10.13053/cys-22-1-2774
19. Zouaghi, A.; Marhbène, L.; Zrigui, M.: A hybrid approach for Arabic word sense disambiguation. Int. J. Comput. Process. Lang. **24**(2), 133–151 (2012). https://doi.org/10.1142/s1793840612400090
20. AlZu'bi, S.; Al-Qatawneh, S.; Alsmirat, M.: Transferable HMM trained matrices for accelerating statistical segmentation time.

In: Fifth International Conference on Social Networks Analysis, Management and Security (SNAMS), pp. 172–176. IEEE (2018)
21. Mohamed, M.A.B.; Mallat, S.; Nahdi, M.A.; Zrigui, M.: Exploring the potential of schemes in building NLP tools for Arabic language. Int. Arab J. Inf. Technol. (IAJIT) **6**(12), 13–19 (2015)
22. Mahmoud, A.; Zrigui, A.; Zrigui, M. A text semantic similarity approach for Arabic paraphrase detection. In: 18th International Conference on Computational Linguistics and Intelligent Text Processing (CICLing), pp. 338–349 (2017). https://doi.org/10.1007/978-3-319-77116-8_25
23. Batita, M.A.; Zrigui, M.: Derivational relations in Arabic Wordnet. In: 9th Global WordNet Conference GWC, pp. 137–144 (2018)
24. Salah, M.H.; Schwab, D.; Blanchon, H.; Zrigui, M.: Système de traduction automatique statistique Anglais-Arabe, pp. 1–8. arXiv:1802.02053v1 [CS.CL] (2018)
25. Amir, S.; Tanasescu, A.; Zighed, D.A.: Sentence similarity based on semantic kernels for intelligent text retrieval. J. Intell. Inf. Syst. **48**(3), 675–689 (2017). https://doi.org/10.1007/s10844-016-0434-3
26. El-Deeb, R.; Al-Zoghby, A.M.; Elmougy, S.: Multi-corpus-based model for measuring the semantic relatedness in short texts (SRST). Arab. J. Sci. Eng. (2018). https://doi.org/10.1007/s13369-018-3232-0
27. Al-Shenak, M.; Nahar, K.; Halwani, H.: AQAS: Arabic question answering system based on SVM, SVD, and LSI. J. Theor. Appl. Inf. Technol. **97**(2), 681–691 (2019)
28. Shehab, A.; Faroun, M.; Rashad, M.: An automatic Arabic essay grading system based on text similarity Algorithms. Int. J. Adv. Comput. Sci. Appl. (IJACSA) **9**(3), 263–268 (2018)
29. Imran, S.; Khan, M.U.G.; Idrees, M.; Muneer, I.; Iqbal, M.M.: An enhanced framework for extrinsic plagiarism avoidance for research articles. Tech. J. **23**(1), 84–92 (2018)
30. Rafiq, M.H.; Razzaq, S.; Kehkashan, T.: UPD: a plagiarism detection tool for Urdu language documents. Int. J. Multidiscip. Sci. Eng. **9**(1), 19–22 (2018)
31. Abooraig, R.; Al-Zu'bi, S.; Kanan, T.; Hawashin, B.; Al Ayoub, M.; Hmeidi, I.: Automatic categorization of Arabic articles based on their political orientation. Dig. Investig. **25**, 24–41 (2018)
32. Issa, F.; Damonte, M.; Cohen, S.B.; Yan, X.; Chang, Y.: Abstract meaning representation for paraphrase detection. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, vol. 1, pp. 442–452 (2018). https://doi.org/10.18653/v1/n18-1041
33. Ezzikouri, H.; Oukessou, M.; Erritali, M.; Madani, Y.: Fuzzy cross language plagiarism detection approach based on semantic similarity and Hadoop MapReduce. In: Recent Advances in Intuitionistic Fuzzy Logic Systems, pp. 181–190. Springer, Cham (2019)
34. Fernando, S.; Stevenson, M.: A semantic similarity approach to paraphrase detection. In: Proceedings of the 11th Annual Research Colloquium of the UK Special Interest Group for Computational Linguistics, pp. 45–52 (2008).
35. Mihalcea, R.; Corley, C.; Strapparava, C.: Corpus-based and knowledge-based measures of text semantic similarity. In: AAAI'06 Proceedings of the 21st National Conference on Artificial Intelligence, vol. 1, pp. 775–780 (2006)
36. Azunre, P.; Corcoran, C.; Dhamani, N.; Gleason, J.; Honke, G.; Sullivan, D.; Ruppel, R.; Verma, S.; Morgan, J.: Semantic classification of tabular datasets via character-level convolutional neural networks, pp. 1–15. arXiv:1901.08456 (2019)
37. Lai, S.; Leung, K.S.; Leung, Y.: SUNNYNLP at SemEval-2018 Task 10: a support-vector-machine-based method for detecting semantic difference using taxonomy and word embedding features. In: Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval-2018), New Orleans, Louisiana, pp. 741–746 (2018). https://doi.org/10.18653/v1/s18-1118
38. He, H.; Wieting, J.; Gimpel, K.; Rao, J.; Lin, J.: UMD-TTIC-UW at SemEval-2016 Task 1: attention-based multi-perspective convolutional neural networks for textual similarity measurement. In: 10th International Workshop on Semantic Evaluation (SemEval-2016), pp. 1103–1108 (2016). https://doi.org/10.18653/v1/s16-1170
39. Salem, A.; Almarimi, A.; Andrejkova, G.: Text dissimilarities predictions using convolutional neural networks and clustering. In: World Symposium on Digital Intelligence for Systems and Machines (DISA), pp. 343–347 (2018)
40. Mahmoud, A.; Zrigui, M.: Artificial method for building monolingual plagiarized Arabic corpus. Computacion y Systemas **22**, 3767–3776 (2018). https://doi.org/10.13053/cys-22-3-3019
41. Alrabiah, M.; Al-Salman, A.; Atwell, E.; Alhelewh, N.: KSUCCA: a key to exploring Arabic historical linguistics. Int. J. Comput. Linguist. (IJCL) **5**(2), 27–36 (2014)
42. Kim, N.; Choi, Y.; Lee, H.; Choi, J.; Kim, S.; Kim, J.; Cho, Y.; Lee, J.: Detection of document modification based on deep neural networks. J. Ambient Intell. Hum. Comput. **9**(4), 1089–1096 (2018). https://doi.org/10.1007/s12652-017-0617-y
43. Sameen, S.; Sharjeel, M.; Nawab, R.M.A.; Rayson, P.; Muneer, I.: Measuring short text reuse for the Urdu language. IEEE Access **6**, 7412–7421 (2018)
44. Mohsen, G.; Al-Ayyoub, M.; Hmeidi, I.; Al-Aiad, A.: On the automatic construction of an Arabic thesaurus. In: 9th International Conference on Information and Communication Systems (ICICS), pp. 231–247 (2018). https://doi.org/10.1109/iacs.2018.8355431
45. Almarwani, N.; Diab, M.: Arabic textual entailment with word embeddings. In: Proceedings of The Third Arabic Natural Language Processing Workshop (WANLP), Valencia, Spain, pp. 185–190 (2017)