ISSN 2007-9737

Artificial Method for Building Monolingual Plagiarized Arabic Corpus

Adnen Mahmoud, Mounir Zrigui

University of Tunis, LaTICE Laboratory,
Tunisia

mahmoud.adnen@gmail.com, mounir.zrigui@fsm.rnu.tn

Abstract. Plagiarism in textual documents is a widespread problem seen the large digital repository existing on the web. Moreover, it is difficult to make evaluation and comparison between solutions because of the lack of plagiarized resources in Arabic language publicly available. In this context, this paper describes automatic construction of a paraphrased corpus in order to deal with these issues and conduct our experiments, as follows: First, we collected a large corpus containing more than 12 million sentences from different resources. Then, we cleaned it up unnecessary data by applying a set of preprocessing techniques. After that, we used word2vec algorithm to create a vocabulary from the collected corpus. It extracted efficiently the semantic relationships between words to exploit. Subsequently, we replaced each word of the source corpus with the most similar vocabulary word based on an index used randomly to eventually obtain a suspect corpus. Different experiments are done. Thus, we varied the dimensions of vectors and window sizes to predict the correct context of words and identify the semantically closest words of the target.

Keywords. Arabic language, automatic creation, data collection, word embedding, paraphrase, plagiarism, semantic analysis.

1 Introduction

Paraphrase allows the reuse of ideas or words in whole or in part from someone else without granting the necessary credit of the source text copied. This phenomenon is of great interest to researchers in order to deal with it. However, Arabic paraphrase detection adds the difficulty of Natural Language Processing (NLP), because of the lack of wide paraphrased corpora in Arabic language available publicly. In this context, the fundamental thought of this paper is to build a monolingual Arabic plagiarized corpus based on an artificial method. The rest of this paper is organized as follows: First, we present the problem statement, in section 2. Then, we expose an overview of related corpora and their properties, in section 3. Thereafter, we describe our proposed method for the construction of a paraphrased corpus in Arabic language, in section 4, and the experiments that we have done, in section 5. Finally, we end this paper by a conclusion and some future works to achieve, in section 6.

2 Problem Statement

The detection of plagiarism in textual documents is difficult because of the large amount of data exchanged in the web. This made unrecognized reuse much more prevalent. Thus, there are different levels of plagiarism, such as: word-to-word plagiarism, the obfuscation of the text content, and the reuse of the main idea independently of the words in the original text [1].

However, plagiarism in Arabic documents has attracted the attention of the research community seen the complexities of Arabic language that they make hard to detect plagiarism in them.

In addition, Arabic language is the language of the Koran and the sacred book of Muslims and the 5th widely used languages in the world [2, 3]. It is spoken by more than 422 million people as a first language and by 250 million people as a second language [4].

Arabic language is characterized by an inflectional, derivational and nonconcatenative morphology compared to other languages, like Latin or English [5]. Among the specificities of

doi: 10.13053/CyS-22-3-3019

Arabic language, which make its analysis more difficult, we mention:

- Arabic script is a Semitic language, which makes it different from other languages. Thus, an Arabic word is a sequence of related entities completely disjoint named sub-words, which in turn formed of one or more characters. It is indeed a morphologically complex language because of the existence of dots, diacritics, stacked letters, etc. [6, 7].

- Words in Arabic language can mean a whole sentence resulting from an agglutination of the grammar. Thus, certain combinations of characters can be written in different ways which increases the ambiguity. It consists of a stem composed by a consonant root and a pattern morpheme, more affixes and enclitics. [8].

- The same text can be in several formats (fully vowel, semi-vowel, or unguarded). On the other hand, the lack of diacritic signs (vowels or diacritic points) in resources increases the problem of lexical and morphological ambiguity. Thus, the grammatical category of the ambiguous word can disambiguate its sense which complicates Arabic (NLP). [9].

- A word may have several possible meanings and senses due to the richness of Arabic language: The possibility of changing words to their synonyms and statements to another structure to the same as active to passive.

- Arabic script is highly derivational and agglutinative [10, 11]: Lexical category (noun, verb, adjective, etc.) in different contexts allows us to have different meanings of words.

To develop and evaluate a paraphrased plagiarism detection system, a large and increasing amount of digital texts is easily and readily available, making it simpler to reuse but difficult to detect. However, we noticed that we need corpora in Arabic language with examples that initiate real plagiarism cases, which make the evaluation and comparison of such solutions possible.

3 Related Work

This section provides an overview of related works that deal with Arabic paraphrase plagiarism corpora construction and their properties.

Plagiarized corpora mean the construction of a source corpus from which passages of texts are extracted and a suspect corpus in which the aforementioned passages are inserted after undergoing obfuscation processing.

Therefore, we distinguish two ways to create corpora, distinguish [12]:

- *Simulated corpora creation* allows to authors or contributors to manually reuse another document based on different obfuscation forms, such as: copy of words, or different word representations with and without diacritics in the case of Arabic language; and paraphrasing of which all kinds of modifications have been applied (restructuring, synonym substitution, etc.) providing that the meaning of the original passage is maintained.
- *Artificial corpora creation* allows the obfuscation strategies applied to fragments extracted automatically from source documents, such as: shuffling words/sentences by changes the order of words, POS-preserving changes of order, synonym substitution, and addition/deletion of words.

Several methods have been proposed to build corpora in different languages, we cite:

The volunteers are encouraged to use their own knowledge of how to paraphrase a piece of text. Thus, in [1], it has been proposed a corpus in Urdu language contained in total 160 documents: 20 source documents have been original Wikipedia papers on well-known personalities and 140 suspicious ones have been manually paraphrased (plagiarized) versions produced by applying different rewriting techniques, such as: synonym replacement, changing intense or grammatical structure, summarizing content, splitting or combining sentence to make new ones.

Also, text reuse occurs when one borrows the text either verbatim or paraphrased from an earlier written text. It has been the main idea in [13].

The process of Urdu Short Text Reuse Corpus (USTRC), creation has been shown. This corpus has been contained 2,684 short Urdu text pairs, manually labeled as verbatim (496), paraphrased (1,329), and independently written (859).

In order to improve the semantic text plagiarism detection, we distinguish also the (PAN-PC) corpus in English language, which is a multi-dialect, expansive scale, open corpus of unoriginality, containing just artificial plagiarism occurrences. This corpus has been used in [14]. Thus, irregular appropriating has been emulated the initiatives of which a human would made to shroud duplicating, muddling through the reordering of the expressions, word substitution, equivalent word, and antonym utilize, erasures, and additions.

The (PAN-PC-10) corpus has been contained 27,073 text records, 15,925 arrangements of suspect reports and 11,148 arrangements of source archives produced utilizing an artificial plagiarism program. Moreover, a single event may be reported in multiple articles in different ways that certain types of noun expressions such as names, dates, and numbers behave as anchors that may not change from article to another.

In [15], a method for paraphrasing automatically from a corpus in Japanese has been described. Thus, expressions that conveyed the same information have been extracted using a simple co-referenced resolver to handle some additional anchors, such as: pronouns. Consequently, the resulting paraphrases have been generalized as models and stored for future use: 195 source documents and 100 paraphrased articles containing different cases of restriction: 106 co-references or restrictions, 37 co-references and restrictions and 32 co-references without restriction.

In contrast, we distinguish a semantic similarity search model for obfuscated plagiarism detection in Marathi language, in [16]. In this system, authors have been identified three datasets: the first two corpora, (PAN-PC-11) and (PAN-PC-10) including 7645 manual paraphrases and 34,310 automatic paraphrases. On the other hand, (PAN-PC-09) has been involved 17,127 artificial cases but no simulated plagiarism cases have been found. For Marathi language, they have been collected some manual and artificial paraphrases for testing.

Also, it has been described an approach to create a monolingual English corpus in (PAN) 2015 competition, in [17]. It has been proposed two different obfuscation methods to create different cases of plagiarism: The first has been an artificial method, which it has been consisted of variety of obfuscation strategies (synonym substitution, random change of order, POS-preserving change of order and addition/deletion).

The second has been a simulated method, in which the (SemEval) dataset has been used for creating the cases of plagiarism using pairs of sentences with their similarity scores. However, we need to develop plagiarism corpora in Arabic language of which standardized evaluation resources have been very beneficial to a wide range of field including information retrieval, (NLP), etc. Thus, few works have been proposed:

In [12], it has been built a plagiarism detection corpus for Arabic. The students have been asked to write an article about the importance of information technology and use the Internet in their sources, especially in the case of a website. Different levels of plagiarism have been applied in the corpus, such as: exact copy, light modification or heavy modification. At the end, the proposed corpus has been composed by 1600 documents of which no plagiarized document has been found.

Likewise, the Corpus of Contemporary Arabic (CCA) has been used, in [18]. This corpus has been contained hundreds of documents in a variety of topics and genres collected from magazines. It has been included in its corpus specifically in the source documents hundreds of documents from Arabic Wikipedia. These documents have been collected manually by selecting documents that match the topics of the suspicious documents.

After, they have been incorporated in the corpus to baffle the detection, and only few cases have been created from them. After that, it has been realized that many of the collected Wikipedia articles (notably biographies) have been contained exact or near exact copies of large passages from the (CCA) documents, including: 1174 documents for the train and 1171 documents for the test.

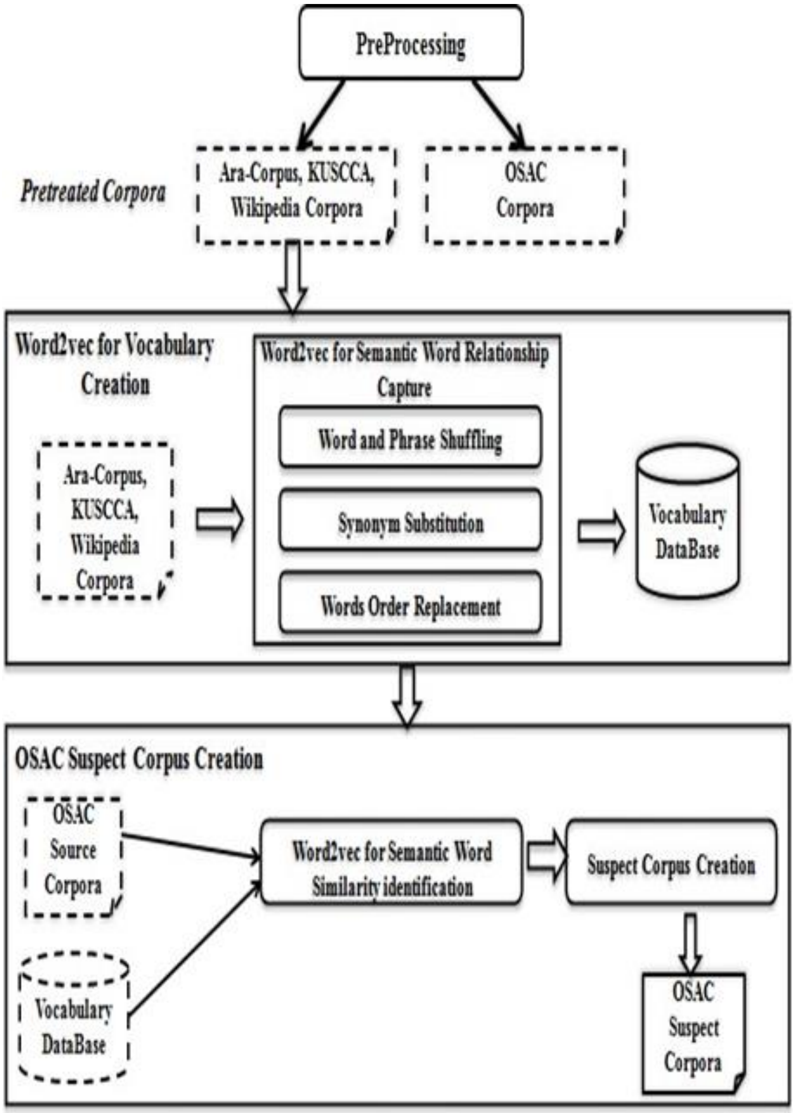
PreProcessing
Pretreated Corpora
Ara-Corpus, KUSCCA, Wikipedia Corpora
OSAC Corpora
Word2vec for Vocabulary Creation
Ara-Corpus, KUSCCA, Wikipedia Corpora
Word2vec for Semantic Word Relationship Capture
Word and Phrase Shuffling
Synonym Substitution
Words Order Replacement
Vocabulary DataBase
OSAC Suspect Corpus Creation
OSAC Source Corpora
Vocabulary DataBase
Word2vec for Semantic Word Similarity identification
Suspect Corpus Creation
OSAC Suspect Corpora

Fig. 1. Proposed construction model of a plagiarized corpus in Arabic language

Table 1. Tokenization forms examples

Forms	Arabic form	English form
Lemma + conjunction	ويخرج	and go out
Lemma + preposition	ليخرج	to get out
Lemma + definitional article	الخروج	the go out
Lemma + possession	خروجه	his go out

Table 2. Collected corpora

Source	Words number
AraCorpus	126.026.301
KSUCCA	48.743.953
Wikipedia	2. 158.904.163
Total	More than 2.3 billion words

4 Proposed Artificial Construction of an Arabic Plagiarized Corpus

In this section, we detail our approach for building a monolingual plagiarized Arabic corpus as shown in figure 1, including the following phases:

First, we collect a large Arabic corpus from different resources.

Then, we convert our datasets into text files (.txt) using UTF-8 encoding text. Thus, it is the standard form used for texts in order to facilitate subsequent processing's.

After that, we apply different preprocessing methods to eliminate unnecessary data within text.

Finally, we try to create our suspect corpus automatically using word2vec algorithm and different random functions.

4.1 Preprocessing

We begin by cleaning our corpora used up unnecessary data by applying a set of pre-processing techniques in order to make subsequent processing easier. Among the operations that we have applied, distinguish:

4.1.1 Corpora Cleaning

Arabic is a rich and highly productive Semitic language, both derived and inflexive which complicated its analysis. Therefore, we applied a set of operations to clean it by removing diacritics, extra white spaces, titles numerations, special characters, and no Arabic letters.

4.1.2 Corpora Tokenization

The Arabic language is characterized by it's cursively and presents various diacritics. Thus, a word is composed by a set of related entities that are in turn formed of one or more characters.

Therefore, considering the complexity of Arabic language morphology and its analysis at the level of sentences and words, we apply the technique of tokenization to extract the words that can designate a whole sentence. This is by separating the lemma and the word representing an assembly element of grammar, such as [19]: conjunctions, prepositions, definitional article or possession.

Here is an example representing some tokenization styles in table 1:

4.2 Train Model

4.2.1 Corpora Collection

We collect Arabic corpora from different resources to accomplish our work, characterized by different changes, such as: lexical, vocabulary and semantic, etc. Our collected corpus is composed by more than 2 billion words as shown in table 2.

Among the corpora that we have collected:

- King Saud University Corpus of Classical Arabic (KSUCCA) represents a set of Arabic texts which are used in various types of computational linguistic research. It is composed by 50 million of words containing Arabic texts classified into 6 genres, such as: religion, linguistics, literature, science, sociology and biography. These genres cover the most of the topics that were popular in that period of time. [20]
- Arabic Corpora Resource (AraCorpus) gives references of Arabic corpora computing linguistics of more than 126 million words. Among the corpora we used: CNN, Thawra, Ahram, Akhbar, Almshaheer, Alquds, Alwatan, Al-watan, aps, Alsharq alawsat, and Attajdid.[1]
- Set of Arabic papers from Wikipedia: composed by more than 400,000 papers and more than 2 billion words among the 295 active language editions, in February 2017.[2]

[1] http://aracorpus.e3rab.com/index.php?content=english

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

Table 3. Test corpus “OSAC”

Document category	Files number
Economics	3102
History	3233
Education & Family	3602
Religious and Fatwas	3171
Sports	2419
Health	2296
Astronomy	557
Low	944
Stories	726
Cooking Recipes	2373

Table 4. Training configuration parameters

Parameters	Values
Vector Size	300
Min_Count	<= 5
Window size	3
Workers	8
Iterations Number	7

4.2.2 Vocabulary Creation

We create our vocabulary using word2vec algorithm in order to enrich the word representation according to its context (the words that surround it). Thus, words are projected to a continuous space of predefined size and words with similar contexts that can induce syntactic and semantic relations [21].

Thus, we use the Skip gram model to predict the possible context of words and overcome the semantic ignorance problem of the data used compared to the “Continuous Bag Of Words CBOW” model.

Formally, given a sequence of words $\{w_1, w_2, \ldots, w_T\}$, we try to maximize the average log probability[3]:

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

where c is the width of the context around each word w_t of which conditional probabilities are defined with Softmax function, as follows:

$$p(w_i|w_j) = \frac{exp(u_{w_i}\ T\ v_{w_j})}{\sum_{l=1}^{V} exp(u_l\ T\ v_{w_j})}, \qquad (2)$$

where V is the number of words in the vocabulary, u_{w_i} is the output representation of $w_i$and v_{w_j} the input representation of w_j.

So, our main idea based on an analogy reasoning to obtain semantic relations between the words to exploit. Moreover, we try to determine the parameters that make our approach efficient doing several experiments by varying the vectors dimensions and the window sizes.

4.3 Test Model

4.3.1 Source Corpus

We use Open Source Arabic Corpora (OSAC) as a source corpus. It includes 22,429 text documents. Each text document belongs to 1 of 10 categories as shown in table 3, such as: economics, history, entertainments, education & family, religious and fatwas, sports, health, astronomy, low, stories, and cooking recipes. [4].

4.3.2. Suspect Corpus Creation

In addition, we are talking of paraphrase when the contents of a text pair are describing the same idea using different text rewrite operations, such as: addition/ deletion of words (or sentences), synonym substitutions, lexical changes, active to passive switching, etc.

Therefore, we propose an artificial paraphrased corpus using the vocabulary that we have built based on word2vec algorithm, as follows:

First, we use random uniform function[4] to identify the rate of plagiarism.

[3] http://cedric.cnam.fr/vertigo/Cours/RCP216/coursFouilleTexte.html

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

ISSN 2007-9737

Table 5. Paraphrased sentence construction example

Source Sentence: خرجت الجماهير المصرية في شوارع القاهرة فور انتهاء المباراة لتحتفل بهذا الفوز “The Egyptian fans went out in the streets of Cairo immediately after the match to celebrate this victory”
Paraphrased sentences according to (vector dimension, window size)
(250, 2)
خرجت جماهير المصرية ضمن الشوارع الاسكندرية بعد بدء المباراة لتحتفل بهذا فوز “The Egyptian fans went out on the streets of Alexandria after the start of the game to celebrate this victory”
(300, 3)
خرجت الجماهير المصرية في شوارع القاهرة عقب انقضاء المباراة لتحتفل بهذا الانتصار “Egyptian fans went out in the roads of Cairo after the match to celebrate the victory”
(300, 4)
خرج جماهير المصرية ضمن شوارع القاهرة اثر انقضاء المباراة لتحتفل بهذا فوز “Egyptian fans took to the streets of Alexandria after the match to celebrate this victory”
(350, 3)
خرجت جماهير المصرية و الشوارع الاسكندرية قبل انتهاء المباراة لتحتفل بهذا فوز “Egyptian fans and the streets of Alexandria went out before the finishing of the game to celebrate this win”

After many experiments that we have done, we fix the plagiarism rate P between 0.45 and 0.75 (Percent of plagiarism between 45% and 75%).

Then, given the total number of words N in a given source document and the rate of plagiarism P to apply, the number of words to replace S is obtained as follows:$S = N \times P$

Since an index of all words of the original document varies between 0 and N- 1, we try to find in which part of the document we will replace the words S according to a random index.

Thus, we use the shuffle function[5] to replace the order of sub-matrices, but their content remains the same.

After that, we try to create the suspect corpus, as follows: First, we extract all words (tokens) from each sentence of the source corpus. Then, we compare the word vectors representation of each token obtained with the words of our vocabulary model that we have built to extract their most similar words. In the case where a given token exits in our vocabulary model, we display a vector containing the most similar words for the current word. Finally, we replace the original word by the most similar word to have thereafter a suspect corpus.

5 Experiments

To conduct our experiments, we use a large collection from different resources (Aracorpus, KSUCCA, and Wikipedia papers), counting more than 2 billion words:

First, we use them to build our vocabulary based on distributed word vectors representation. After that, we use the vocabulary that we have built as a training model. The goal is to exploit more concepts to handle the rich morphology of Arabic language. Then, we evaluate the quality of our proposed methodology using (OSAC), corpus as a source corpus to generate thereafter the plagiarized corpus.

5.1 Train Model

To perform the performance of the paraphrased corpus creation, we used an analogy reasoning using word2vec algorithm based on Skip gram model.

Thus, our main idea is to learn the relationships between words according to their context in the level of vocabulary and suspect corpus creations.

In this context, we experiment with different parameters to identify the semantic relations between words to exploit, as follows: we have varied the parameters used in order to perform the suspect corpus creation, as follows:

- Different vector dimensions, such as: 100, 250, 300, 350, and 500;

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

– Different window sizes to say the correct context of words, such as: [1,…, 10].

Finally, the best parameters that affect the resulting vectors which they performed our experiments are: 300 as a vector dimension and 3 as the window size, as shown in the following table 4 representing the final training configuration parameters that we have used to construct our proposed model: Where: vector size is the dimensionality of the word vectors; Min_Count is the words that have a total frequency less than 5 will be ignored; window size is the maximum distance between the current and predicted word within a sentence; workers is the train model using many worker threads; and iterations number is the number of iterations or epochs over the corpus.

5.2 Test Model

Our main goal is how to build an efficient corpus to conduct our experiments for paraphrase detection. Thus, our proposed methodology allows replacing the words of the (OSAC), source corpus with the vocabulary words most similar using an index randomly to eventually obtain the (OSAC), suspect corpus.

So, among the advantages of our proposed model, we mention:

– The use of Skip-gram model is an efficient method for learning high quality of distributed vectors representation. This can capture a large number of precise syntactic and semantic relationships for better performance.

– Moreover, the construction of suspect corpus is done in an automatic way of which we distinguish different types of obfuscations, such as: total copy of the source document of which there is no obfuscation; shuffling of source words or sentences at random by changing their order randomly; and replacing source words by their most similar words if they exist in the vocabulary.

Here is an example in table 5 illustrating our proposed automatic construction of a paraphrased corpus by dealing with the cases of creating a suspect sentence from a source sentence.

After several experiments of more than 1000 words according to different vector dimensions and window sizes, we conclude that the parameters that help us to build the best paraphrased corpus are: 300 as a vector dimension and 3 as the window size.

Table 6. Paraphrased words construction example

Word	**Similar words according to (Vector dimension, Window size)**			
	(250, 2)	**(300, 3)**	**(300, 4)**	**(350, 3)**
“في”: in	“ضمن”: on 0.516	“في”: in 0.593	“ضمن”: to 0.542	“و”: and 0.532
“فور”: immediately ”	“بعد”: after 0.659	“عقب”: after 0.789	“اثر”: after 0.667	“قبل”: before 0.638
“انتهاء” : after/ finishing	“بدء” :the start 0.651	“انقضاء”: the end 0.688	“انقضاء”: the end 0.577	“انتهاء” : the finishing 0.58

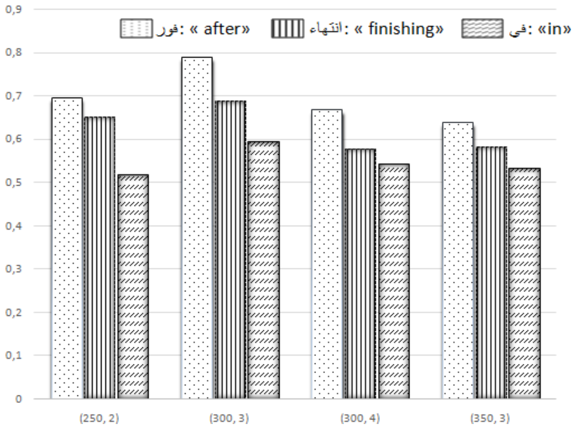

Fig. 2. Examples of test configuration parameters

Here is an example in table 6 and figure 2 illustrating some paraphrased words according to the parameters that we have concluded and their most similar words found based on Cosine similarity measure integrated in word2vec algorithm:

6 Conclusion

Paraphrase plagiarism is a significant problem and researches show that is hard to detect. Also, the lack of resources representing different cases of plagiarism especially in Arabic language complicates the evaluation of our solutions.

That is why we proposed an artificial based method for building a plagiarized corpus to deal with it. Our main purpose is based on word2vec algorithm to create a vocabulary from the collected corpus of which it was efficient to extract semantic relationships between words to exploit. Moreover, we used it to replace each word from the source corpus with the most similar vocabulary word randomly to eventually obtain a suspect corpus. Experiments are shown that our proposed plagiarized corpus represent different cases of plagiarism, such as: total copy, source word/sentence shuffling, and synonym substitution.

For future work, we try to improve the performance of our proposed method using Part Of Speech Tagging (POS-Tag) technique. It allows providing morphological and syntactic annotations based on the grammatical class of the word.

References

1. **Sharjeel, M., Rayson, P., Muhammad, R., & Nawab, A. (2016).** UPPC - Urdu Paraphrase Plagiarism Corpus. *Proceedings of Tenth International Conference on Language Resources and Evaluation (LREC)*, pp. 1832–1836.
2. **Mahmoud, A. & Zrigui, M. (2017).** Semantic similarity analysis for paraphrase identification in Arabic texts. *Proceedings of the 31st Pacific Asia Conference on Language, Information and Computation**,** Philippine (PACLIC),* Vol. 31, pp. 274–281.
3. **Zrigui, S., Zouaghi, A., Ayadi, R., Zrigui, S., & Zrigui, M. (2016).** ISAO: An intelligent system of opinion analysis. *Research in Computing Science,* Vol. 110, pp. 21–30.
4. **Saad, M. K. & Ashour, W. (2010).** OSAC: Open Source Arabic Corpora. *Proceedings of 6th International Conference on Electrical and Computer Systems (EECS'10), Lefke, North Cyprus,* pp. 1–6.
5. **Batita, M. A. & Zrigui, M. (2018).** Derivational relations in Arabic Wordnet. *The 9th Global WordNet Conference GWC*, pp. 137–144.
6. **Mansouri, S., Charhad, M., & Zrigui, M. (2018).** A heuristic approach to detect and localize text in Arabic news video. *Computación y Sistemas*, Vol. 23, No.1, pp. 75–82.
7. **Hkiri, E., Mallat, S., & Zrigui, M. (2017).** Arabic-English text translation leveraging hybrid NER. *Proceedings of the 31st Pacific Asia Conference on Language, Information and Computation***,** Vol. 31*,* pp. 124–131*.*
8. **Boudhief, A., Maraoui, M., & Zrigui, M. (2014).** Elaboration of a model for an indexed base for teaching Arabic language to disabled people. *6th International Conference on (CIST)*, pp. 110–116.
9. **Meddeb, O., Maraoui, M., & Aljawarneh, S. (2017).** Hybrid modelling of an off line Arabic handwriting recognition system: results and evaluation. *International Journal Intelligent Enterprise (IJIE),* Vol. 4, No. 1/2, pp.168–189.
10. **Mahmoud, A., Zrigui, A., & Zrigui, M. (2017).** A text semantic similarity approach for Arabic paraphrase detection. *International Conference on Computational Linguistics and Intelligent Text Processing CICLing*.
11. **Ben-Mohamed, M. A., Mallat, S., Nahdi, M. A., & Zrigui, M. (2015).** Exploring the potential of schemes in building NLP tools for Arabic language. *International Arab Journal of Information Technology (IAJIT),* Vol. 6, No. 12, pp.13–19.
12. **Siddiqui, M. A., Khan, I. H., Jambi, K. M., Elhaj, S. O., & Bagais, A. (2014).** Developing an Arabic plagiarism detection corpus. *Computer Science & Information Technology (CS & IT)*, pp. 261–269.
13. **Sameen, S., Sharjeel, M., Nawab, R. M. A, Rayson, P., & Muneer, I. (2017).** Measuring short text reuse for the Urdu language. *Language Resources & Evaluation*, Vol. 51, No. 3, pp. 777–803.
14. **Osman, A. H. & Barukab, O. M. (2017).** SVM significant role selection method for improving semantic text plagiarism detection. *International Journal of Advanced and Applied Sciences*, Vol. 4, No. 8, pp. 112–122.
15. **Shinyama, Y. & Sekine, S. (2014).** Paraphrase acquisition for information extraction. *Artificial Intelligence Review,* Vol. 42, No. 4, pp. 851–894.
16. **Shenoy, N. & Potey, M. A. (2016).** Semantic similarity search model for obfuscated plagiarism detection in Marathi language using Fuzzy and Naïve Bayes approaches IOSR. *Journal of Computer Engineering,* Vol. 18, No. 3, pp. 83–88.
17. **Mohtaj, S., Asghari, H., & Zarrabi, V. (2015).** Developing monolingual English corpus for plagiarism detection using human annotated paraphrase corpus. *Notebook for PAN at CLEF´15.*

18. **Bensalem, I., Boukhalfa, I., Rosso, P., Abouenour, L., Darwish, K., & Shikhi, S. (2015).** Overview of the AraPlagDet PAN@FIRE2015 Shared Task on Arabic plagiarism detection. *PAN´15*, pp. 111–122.
19. **Taamallah, S. B. (2012).** *Prétraitement de données et création d'un segmenteur de l'arabe pour un système de traduction probabiliste vers le français*. Université Stendhal Grenoble.
20. **Alrabiah, M., Al-Salman, A., Atwell, E., & Alhelewh, N. (2014).** KSUCCA: A key to exploring Arabic historical linguistics. *International Journal of Computational Linguistics (IJCL)*, Vol. 5, No. 2, pp. 27–36.
21. **Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013).** Distributed representations of words and phrases and their compositionality*. Advances in Neural Information Processing Systems,* Vol. 26, pp. 3111–3119.

Article received on 16/12/2017; accepted on 03/03/2018. Corresponding author is Adnen Mahmoud.