

Original Research Article

Detecting Plagiarism in Arabic E-Learning Using Text mining

ABSTRACT

Being a growing problem, plagiarism is generally defined as a “ literary theft ” and an “ academic dishonesty ” in the literature, and it is really has to be well-informed on this topic to prevent the problem and stick to the ethical principles. With the hug of the information on WWW and digital libraries, Plagiarism became one of the most important issues for universities, schools and researcher’s fields. It is so easy through the internet and due to using advanced search engine to find documents or journals by students. So plagiarism is a global problem, which occurs in many different areas of our life. It is pivotal to mention here that detecting plagiarism is a challenging task.

E-Learning systems in Arab countries, On the other hand, necessitate technology for the purpose of detecting plagiarism in Arabic. Although search engines such as Google can be utilized, there would be boring efforts to copy some sentences and paste them into the search engine to find similar resources. For that reason, developing Arabic plagiarism detection tool for e-learning systems facilitate and accelerate the process since plagiarism can be detected and highlighted automatically, and one only needs to submit the document to the system. Therefore, this paper propose An effective web-enabled system for Arabic plagiarism detection Using Text mining , ZPLAG, that can be integrated with e-learning systems to judge students’ assignments, papers and dissertations. And presents its experimental results.

KEYWORDS: *Arabic, E-Learning, Plagiarism Detection.*

1- Introduction

According to www.plagiarism.org, plagiarism is any of the following activities: Turning in someone else's work as your own, copying words or ideas from someone else without giving credit, failing to put a quotation in quotation marks, giving incorrect information about the source of a quotation.

There are many different forms of plagiarism; Plagiarism at schools can be a highly de-motivating factor for teachers and also for students. If plagiarism is not addressed sufficiently, plagiarists could gain undeserved advantage, e.g. more marks for their assignments with less effort. [1]

Plagiarized document detection plays important roles in many applications, such as file management, copyright protection, and plagiarism prevention. [2]. Plagiarism can take one of the popular types such as copying of the whole or some parts of the document, rewording same content in different words, using others’ ideas or referencing the work to incorrect or non-existing sources [3]. Other ways of plagiarism include translated plagiarism wherein the content is translated and used without referencing the original work, artistic plagiarism in which different media such as images and videos are used to present other’s work without proper citation [4]

E-Learning systems have gained popularity as a valuable educational environment in the past few decades. Recent advances in networks, multimedia and information technology have contributed to the attractiveness of e-learning systems. Recently, Saudi Arabia (SA) has emerged the use of e-learning systems in universities and

schools. According to a study conducted by Madar Research [5], e-learning industry in SA is projected to grow up significantly in the next few years. It is expected for SA universities to switch from traditional learning to e-learning systems within the next few years [6].

However, the ease of teaching and learning through e-learning systems congregates the difficulty of ensuring intellectual property of students' submitted work. With the ease of using the Internet to access vast amount of information, the problem of plagiarism (the use of other's work or ideas without proper citation) has horribly increased. According to some studies about academic dishonesty, at least 10% of students' work could be plagiarized in USA, Australia and UK universities [7]. Therefore, detecting and deterring plagiarism in the process of e-learning is considered crucial especially in the process of evaluating students' submitted work. Plagiarism has revealed its effects on poor academic results [8]. And finally source code plagiarism (also called code clone) which can be defined as the reuse of the source code without permission or citation. All these practices of plagiarism have negative impact on the learning process. Thus, how can we ensure dealing with plagiarism in e-learning systems and how is plagiarism going to be detected and dealt with. It is a critical issue that needs solutions by computer scientists. In this paper, we presented the experimental results of a novel web enabled tool specific for plagiarism detection in Arabic documents.

2- Related work

Most of the work in document plagiarism has been done for academic purpose. Detecting plagiarism is important to judge and mark students' work especially for postgraduates who are strictly prohibited from cheating, rewording, rephrasing, or restating without referencing. In this regard, numerous plagiarism detection systems have been developed. Most of these systems use plagiarism techniques known as similarity detection techniques, which create special "fingerprints" for collection files, including metrics, such as average line length, file size, average number of commas per line. The files with close fingerprints are treated as similar. Clearly, small fingerprint records can be compared rapidly, but this technique is now considered unreliable and rarely used nowadays. These systems will be discussed as follows.

1. Turnitin is the global leader in evaluating and improving student writing. The company's cloud-based service for originality checking, online grading and peer review saves instructors time and provides rich feedback to students. One of the most widely distributed educational applications in the world, Turnitin is used by more than 10,000 institutions in 126 countries to manage the submission, tracking and evaluation of student papers online. Institutions license Turnitin on an annual basis. The institutions are encouraged to communicate with students about their use of Turnitin and how their academic integrity policies work. An instructor sets up a class and an assignment in the Turnitin service. Students or instructors then submit papers to Turnitin via file upload or copy-and-paste[9].

2. APD (Arabic Plagiarism Detection) tool use the Internet to help professors and teachers in e-learning systems identify stolen intellectual property by utilizing Google API to find similar documents on the web [10]. The typical workflow in APD paradigm has two major steps. The first step, students submit their assignments in Arabic to the system, which in turn will be stored into reports database. The second step, the teacher triggers APD tool via a user interface to check the assignments for plagiarism. Then, the tool will compare the documents against the intra corpus collection which probably contains the previous assignments. Moreover, APD tool searches the web to give similar resources as well. An automatic report will be generated that contains highlighted plagiarized parts and a list of similar resources ranked from highest to lowest. [11]

3- *The Proposed System (ZPLAG)*

A plagiarism detection system for natural languages should satisfy the following properties:

- Insensitivity to punctuation, extra whitespaces, capitalization, etc.
- Insensitivity to small matches (a match should be large enough to imply plagiarism).
- Insensitivity to permutations of the document content.

Our plagiarism detection tool, ZPLAG, is built around a content-based method. It fulfills the three properties. The first property is handled by a preprocessing of any input text, including tokenization, stop-word removal, rooting and synonym replacement. ZPLAG is based on fingerprinting N-grams. The second property is satisfied if “N” is sufficiently long to ignore common idioms of Arabic language. The third property is demonstrated by the performance results on the data set “Structure change”.

The main architecture of ZPLAG is described in Fig1. Its most important design issues are related to:

- Preparation: text editor, check language, check spelling, and check grammar.
- Preprocessing: tokenization, stop-word removal, rooting, and synonym replacement.
- Fingerprinting: make use of n-grams, where n is a parameter chosen by the user.
- Document representation: for each document, create a document tree structure that describes its internal representation.
- Selection of a similarity metric: use of a similarity metric to find the longest match of two hash strings.

The proposed system architecture can be explained as follows:

First, the faculty member has to choose a subject to make a quiz or a paper request, then he sends the quiz or the request for the paper or the thesis to the students. After that each student sends his answer to the faculty member from his own home page of the site in a document file. After the faculty member receives all students answers, he will send it to plagiarism detection tool (PDT). That tool, first, check the documents language. If the document is written in Arabic, the tool will choose Arabic process to detect plagiarism. If the document is written in English, the tool will choose English process to detect plagiarism. The tool, then, makes a plagiarism report, and sends the report to the faculty as well as a copy of the report to the student.

Furthermore, the faculty members can send their papers or books to the PDT. The PDT does the same with the student's work. It checks the documents written language, then chooses the correct process, and finally, makes the plagiarism report, but it sends the report to the faculty members only.

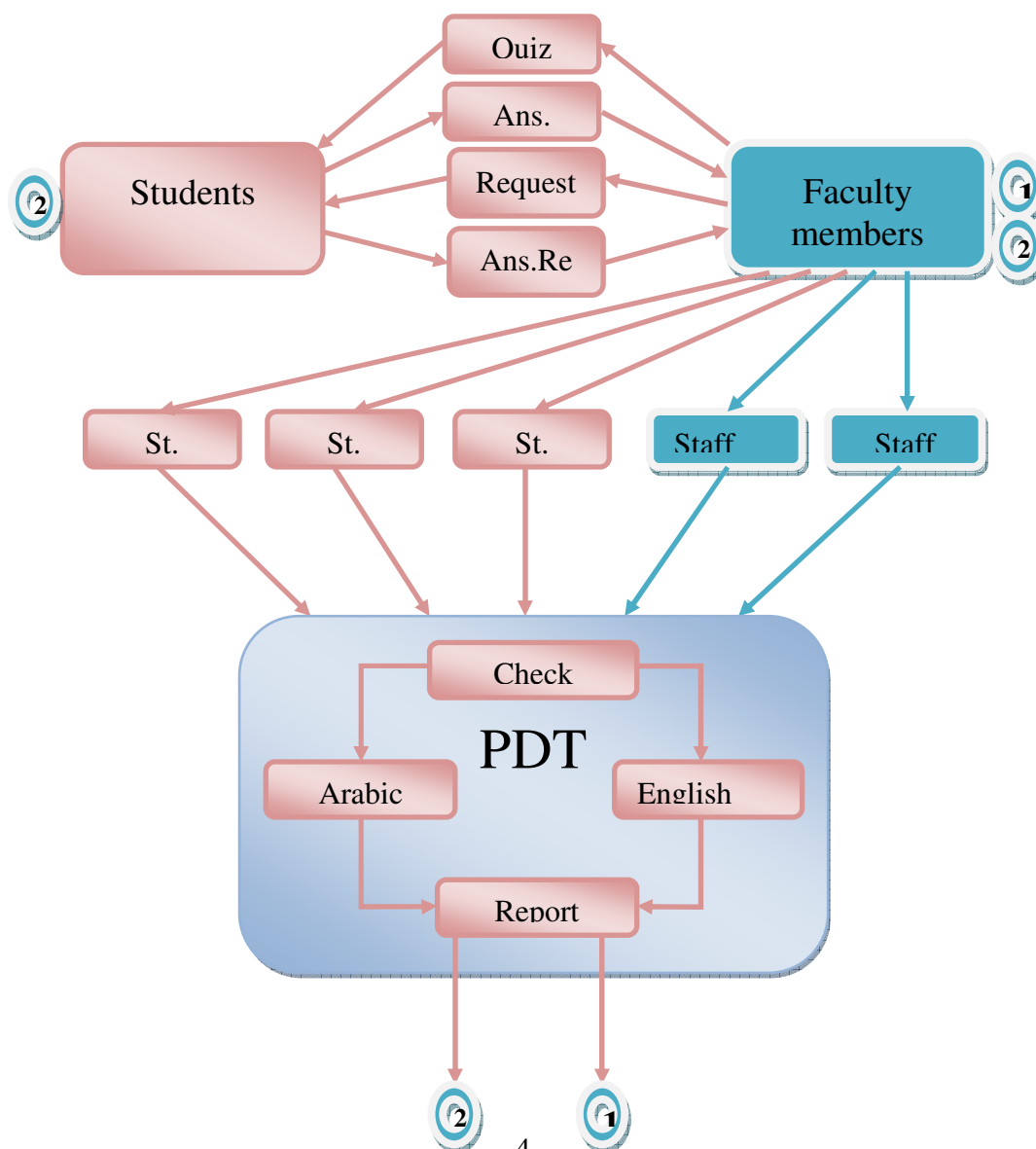


Figure 1 proposed system Architecture

4- Experimental Results

We implemented a prototype of ZPLAG and evaluated its performance on a handmade data test set of 100 Arabic documents of about 500 words each. We extracted 20 documents from different books available on Saudi Digital Library (SDL) [<http://sdl.edu.sa/SDLPortal/AR/Publishers.aspx>]. We generated 3 data sets from the original documents as follows, and the results were obtained on only 10 documents and summarized in table 1 and table 2 where table 1 summarize the Mean (Precision) and table 2 Mean(Recall) for the three types of the data set:

- Data set: *Synonym*

10 candidate documents were generated from each original document by replacing randomly 50% of the total number of words in each document with one of their synonyms. Stop-words were not considered.

- Data set: *Structure change*

10 candidate documents were generated from each original document by changing the structure of randomly selected sentences. The number of generated sentences represents 50% of the total number of sentences.

- Data set: *All data*

10 candidate documents were generated from each original document by copying randomly selected sentences (40% of the total number of sentences), replacing selected words with one of their synonyms (20% of the total number of words), and changing the structure of selected sentences (40% of the total number of sentences).

The data sets Synonym and Structure change were used to evaluate the performance of ZPLAG in detecting hidden plagiarism. The data set All data served to measure ZPLAG's overall performance in detecting hidden plagiarism and exact copy of parts of texts. Three variants of ZPLAG were tested to measure the impact of Stop-Word Removal, rooting, and synonym replacement:

- SWR: only stop-word removal is applied to the input texts.
- SWR+Rooting: stop-word removal and rooting are applied to the input texts.
- SWR+Rooting+Synonym: stop-word removal, rooting, and synonym replacement are applied to the input texts.

The chunk parameter was set to 3. The document threshold DocThreshold was set to 0.1 assuming that documents describing different subjects have an intersection less than 10% of the minimum document size. The paragraph threshold ParThreshold, sentence threshold SenThreshold, and similarity threshold SimilarityThreshold were set to 0.2, 0.1, and 0.5, respectively. Performance results were measured using Recall (1) and Precision (2) metrics.

$$\text{Recall} = \frac{\text{number of plagiarized sequences identified}}{\text{total number of sequences identified}} \times 100 \quad (1)$$

$$\text{Precision} = \frac{\text{number of plagiarized sequences}}{\text{total number of sequences}} \times 100 \quad (2)$$

Figures 3 and 4 show respective mean precision (Mean (precision)) and mean recall (Mean(recall)) obtained by ZPLAG's variants on the 3 data sets. The results obtained can be summarized as follows:

- SWR does not detect hidden plagiarism (synonym replacement and structure change). Its overall performance on all data sets is weak (Mean (precision)=53%, Mean(recall) =37%).

- SWR+Rooting does not detect synonym exchanges, but it can identify changed sentence structure with high precision and recall(Mean (precision)=95%, Mean(recall) =72%). This shows that reducing words to their root can enhance the performance of the plagiarism detection.

- SWR+Rooting+Synonym is the best performing ZPLAG's variant achieving Mean (precision)=97%, and Mean(recall) =94%. Synonym replacement is detected with Mean (precision)=96%, while sentence structure change is detected with Mean (precision)=93% .

	SWR	SWR+Rooting	SWR+Rooting+Synonym
Synonym	--	--	96
Structure change	--	93	95
All data	53	95	97

Table 1 summarize the Mean (Precision) for the three types of the data set

	SWR	SWR+Rooting	SWR+Rooting+Synonym
Synonym	--	--	93
Structure change	--	87	91
All data	37	72	94

Table 2 summarize the Mean (Recall) for the three types of the data set

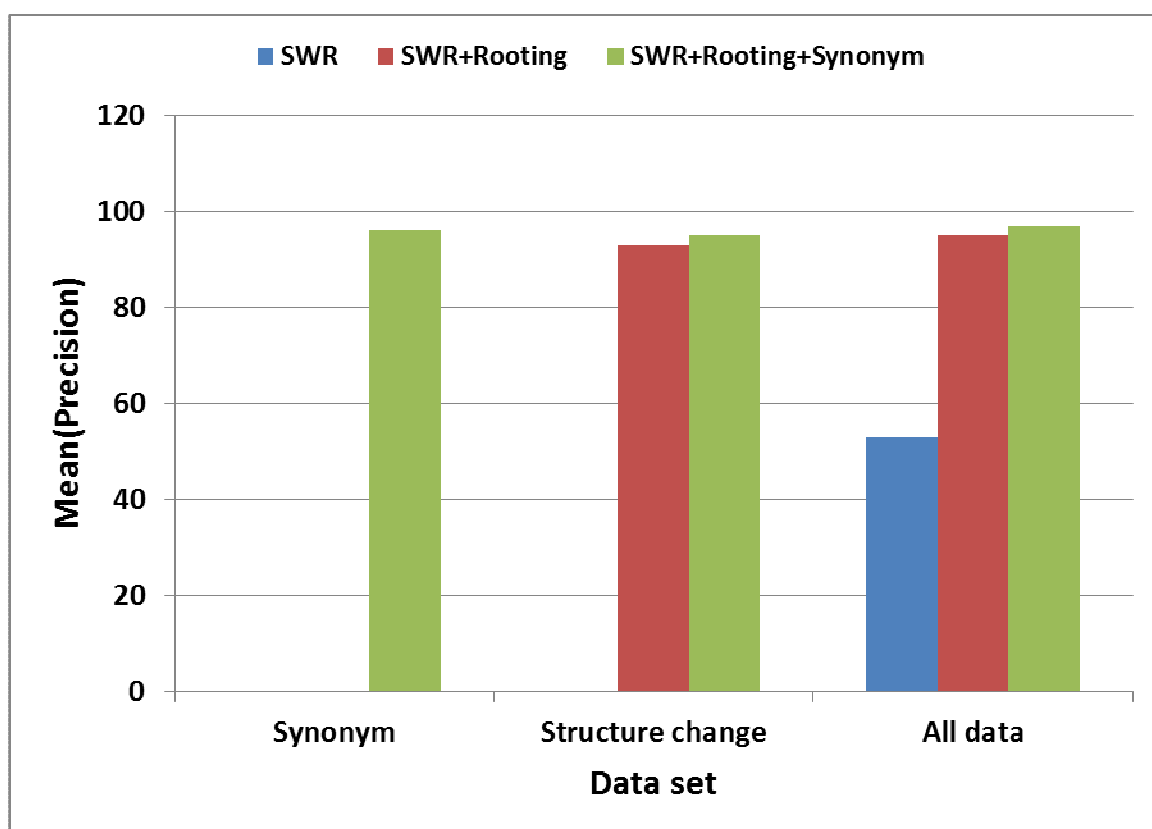


Fig. 3: Mean precision of ZPLAG for each data set

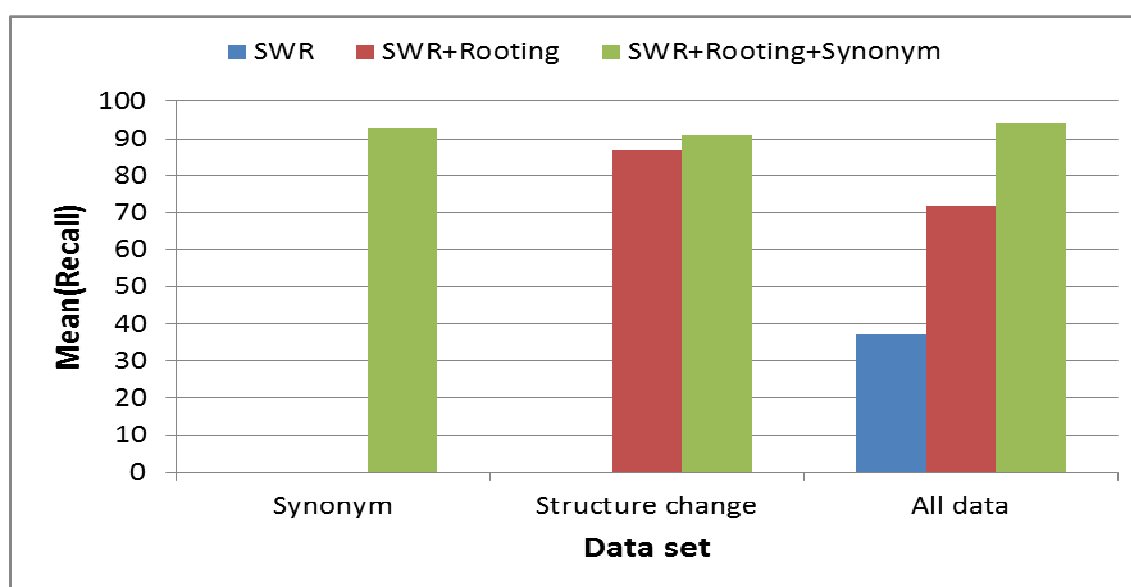


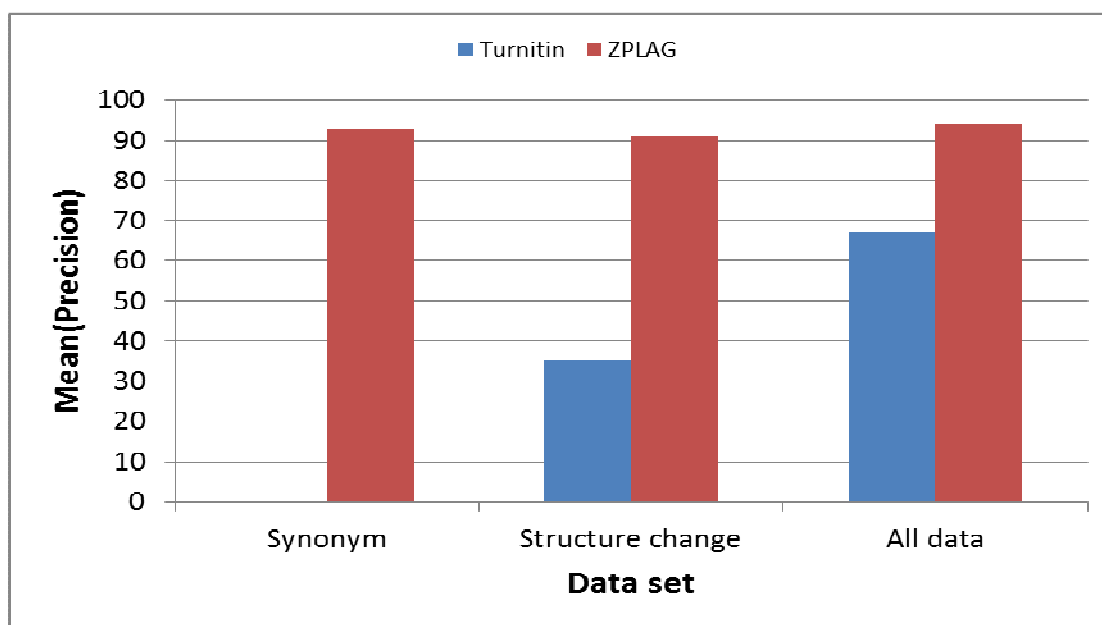
Fig. 4: Mean recall of ZPLAG for each data set

Turnitin was used as a comparative baseline for ZPLAG. It was set to exclude small matches by less than 1%. The performance results of Turnitin and ZPLAG are summarized in table 3. Figure 5 shows the Mean(precision) for ZPLAG and Turnitin for each data set. Turnitin was not able to detect any synonym replacement, but its performance is close to ZPLAG's one in detecting changes in text structure: Mean(precision)=91% for ZPLAG and Mean(precision)=35% for Turnitin. Overall, ZPLAG outperformed Turnitin: Mean(precision)=90% for ZPLAG and Mean(precision)=67% for Turnitin. Although Turnitin is worldwide used, its results for detecting similarities in our data sets are not competitive. This indicates that language-independent

182 tools could be actually inefficient on specific languages, such as Arabic. Table 3 summarize the Mean (precision)
183 for ZPLAG and Turnitin.

	Turnitin	ZPLAG
Synonym	--	93
Structure change	35	91
All data	67	94

184 Table 3 summarize the Mean(precision) for ZPLAG and Turnitin



185
186 Fig. 5: Mean (precision) for ZPLAG and Turnitin

187 Table4 reports comparison results of ZPLAG (SWR+Rooting variant) and APD. It shows Mean (Recall) its
188 standard deviation σ (Recall), Mean(Precision) and its standard deviation σ (Precision). The results were obtained on
189 only 50 documents . The results of APD are close to those of ZPLAG variant without synonym processing.

190 ZPLAG's performance is dependent on Khoja's stemmer and synonyms retrieved from local database.
191 According to the comparative evaluation study of Arabic language morphological analyzers and stemmers [14],
192 Khoja's stemmer achieves the highest accuracy then the tri-literal root extraction algorithm [15] and the Buckwalter
193 morphological analyzer [16]. So, we do not expect to increase the performance of ZPLAG by using other stemmers.
194 However, using other synonym databases might impact its performance.

195 Table 4: Comparison results of ZPLAG (SWR+Rooting) and APD

	ZPLAG	APD
Mean(Recall) %	99	84.8
σ (Recall) %	5	--
Mean(Precision) %	95	90
σ (Precision) %	2	--

196
197
198
199
200

5- Conclusion

We have presented and tested experimentally ZPLAG, a prototype of a plagiarism detector for Arabic documents in which some hidden forms of plagiarism can be detected, such as sentence structure change and synonym replacement. We have described its framework.

Also, In conclusion, integrating a plagiarism detection tool into e-learning systems is significant and important. There is a great demand to insure the intellectual property for the students' submitted work in Arabic because (i) e-learning industry has increased in Saudi Arabia and other Arab countries and (ii) there is no available tool for Arabic plagiarism detection. This paper shed light on this issue and proposed an effective paradigm of document submission in e-learning system, plagiarism detection process and the integration of both. The results show that ZPLAG system has excellent deal with Arabic scripts and allows students to submit assignments to their teachers in e-classrooms. The teacher, in turn, can retrieve the students' assignments in one of his/her classes and view a report that highlights the plagiarized parts in each submitted assignment. The great deal of ZPLAG system is not only detecting and deterring plagiarism abut also it helps educating students about the importance of the originality by citing the original references. The future work will focus on enhancing and adding more options in this tool.

REFERENCES

1. A. M. Ali, H. M. Abdulla, and V. Snasel "Survey of Plagiarism Detection Methods", Fifth Asia Modelling Symposium, 2011.
2. F. Sanchez-Vega, E. Villatoro-Tello , M. Montes-y, L. Villase, P. Rosso," Determining and characterizing the reused text for plagiarism detection" , Contents lists available at SciVerse ScienceDirect , Expert Systems with Applications 2013.
3. G. Oberreuter, and J. D. Velásquez, "Text mining applied to plagiarism detection: The use of words for detecting deviations in the writing style", Contents lists available at SciVerse ScienceDirect, Expert Systems with Applications 2013.
4. L. Romans, G. Vita, and G. Janis, "Computer-based plagiarism detection methods and tools: an overview", the 2007 international conference on Computer systems and technologies. 2007.
5. AMEINFO. "Saudi Arabia's e-Learning industry". [News Article], cited January, 2014.
6. U.W. SAUDI ARABIA, "E-learning education shakeup.", June 2013 , Available from: <http://www.universityworldnews.com/article.php?story=20080529145753433>.
7. [7] C. Lyon, R. Barrett, and J. Malcolm, Plagiarism is Easy, but also Easy To Detect. Plagiarism: Cross-Disciplinary Studies in Plagiarism, Fabrication, and Falsification. , 2006.
8. R.H. McCuen, "The Plagiarism Decision Process: The Role of Pressure and Rationalization. Education", IEEE Transactions on, 2008.
9. L. Chao, C. Chen, and J. Han "GPLAG: detection of software plagiarism by program dependence graph analysis", the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. 2006.

- 238 10. C.J. Neill, and G. Shanmuganthan, “A Web-enabled plagiarism detection tool”, IT Professional, 2004.
- 239 11. S. M. Alzahrani, and N. Salim, “Statement-based fuzzy-set IR versus fingerprints matching for plagiarism
- 240 detection in Arabic documents,” In Proc. of the 5th Postgraduate Annual Research Seminar, Malaysia, 2009.
- 241 12. A. Farghaly, and K. Shaalan ,”Arabic natural language processing: challenges and solutions “. ACM
- 242 Transactions on Asian Language Information Processing, 2009.
- 243 13. K. Seki, W. Tsukahara, and T. Okamoto,”System development and practice of e-learning in graduate school”.
- 244 in Advanced Learning Technologies, 2005. ICALT 2005. Fifth IEEE International Conference on. 2005.
- 245 14. M. Sawalha, and E. Atwell “Comparative evaluation of Arabic language morphological analysers and
- 246 stemmers”. In: Proceedings of 22nd International Conference on Computational Linguistics (COLING 2008),
- 247 Manchester, UK, Aug. 2008.
- 248 15. H. Al-Serhan, R. Al Shalabi, and G. Kannan ,”New approach for extracting Arabic roots “. In: Proceedings of
- 249 the International Arab Conference on Information Technology (ACIT’2003), Portland, Oregon, USA, 2003.
- 250 16. T. Buckwalter ,”Issues in Arabic orthography and morphology analysis”. In: Proceedings of the Workshop on
- 251 Computational Approaches to Arabic Script-based Languages (Semitic’04), Geneva, Switzerland, 2004.