Sudan University of Science and Technology
(SUST)

College of Graduate Studies

A Model and Framework for Plagiarism Detection in Arabic Documents in Arabic Language

نموذج وإطار للكشف عن الانتحال في الوثائق باللغة العربية

A Thesis Submitted for the fulfillment of the requirement of the degree on PhD in Computer Science

By:
Yahya Ali Abdelrahman Ali

Supervisor:
Professor.Izzeldin Mohamed Osman

August 2018

DEDICATION

This dissertation is dedicated to my Parents Prof.Dr Ali Abedelrahan Ali Father , Samaria Ahmed Abdellah Mather, and my Souad Mohammed Fadulalmula wife who helped and encouraged me to realize my dreams and finish my dissertation.

I would like to thanks To my colleagues in Najran university college of computer science and information system , for helping me in all things, thanks for your valuable support.

To my lovely kids Mohammed, and Ali.

ACKNOWLEDGMENTS

First Praise be to Allah, the Most Gracious and Most Merciful Who has created the mankind with knowledge, wisdom and power. First and foremost I would like to express my thanks to Almighty ALLAH on the successful completion of this research work and thesis.

I would like to express my special deep appreciation and thanks to my advisor professor Dr.Izzeldin Mohamed Osman, and Dr. Ahmed Khaild for helping me and giving me many valuable comments and guidelines during my study. I also want to thank them for your innovative and brilliant comments and suggestions.

I extend my sincere thanks and gratitude to my beloved wife Souad Mohammed Fadulalmula for her help, support, and participation.

I also extend my thanks and appreciation to Dr. Mr. Mohammed Al Bashir, Department of Computer Science, King Saud University with the help of his support

Finally, I am also thankful to my colleagues at Najran University College of computer Science and Information System for their help, support

ABSTRACT

Plagiarism has become an infamous problem in the global academic community. Detection of plagiarism in Arabic documents is particularly a challenging task due to the complexity of the structure of the language. This dissertation provides a model and framework for detection of plagiarism in Arabic documents, which is based on a logical representation of a document as paragraphs, sentences, and words. The main purpose of this research is to develop and implement the Arabic Documents Plagiarism Detection Model "ADPDM" which is based on the model that is capable in detection of plagiarism in Arabic documents and search mechanism for the similar candidate documents within the corpus collection. Through developing pre-processing method including stop word removal, stemming and rooting. The implementation is constructe around a content-based method consisting mainly in fingerprinting the texts according to Arabic language specificity and comparing their logical representations by using Heuristic algorithms. We have introduced a plagiarism detection tool for Arabic language by using the Brian Kernighan and Dennis Ritchie (BKDR) hash function for chunk (3-gram) hashing. The second goal of the logical document representation is to save computation time by avoiding unnecessary comparisons. For that reason, we have defined a heuristic algorithm for each level in the tree: document level, paragraph level, and sentence level. We measure it using the Longest Common Substring (LCS) metric. The ADPDM system for detecting plagiarism in electronic resources for Arabic documents were tested and evaluated using a set of the corpora used in this study. It has 100 documents, 90% of the documents were collected from AraPlagDet (Arabic Plagiarism Detection) web-site divided in three categories dataset1 (Small) , Dataset2 (medium) and dataset3 (Large) , and 10% of the documents were collected from the Decision Support System (DSS) document. The original documents has builded randomly replces and were constructed with different degrees of plagiarism Named dataset4. In this study, preliminary experiments were conudacted using our tool ADPDM and WCopyFind. The result shows that percentages of dateset1 is 14% plagiarize detection during 501 second where WCopyFind is detected 0% in 135 second, in dataset2 shows 8% in 1374 second where WCopyFind is detected 0% in 475 second. As well as dataset3, shows 18% in 1430 second where WCopyFind is detected 6.33% in 271 second, while dataset4 is detected 94% in1682.79 second where WCopyFind find out 81.44% in 357 second. The main conclusion that ADPDM is the best result handled plagiarism detection while it is weak in the time taken and WCopyFind it is weak to handled plagiarism detection while it best in the time taken. Filnaly, the experimental results shows perfect performance of ADPDM as it achieved a Recall value represents 0.780351, with Precision of 0.994264 and F- Measure 0.865688.

المستخلـــــــص

أصبح الانتحال مشكلة سيئة السمعة في المجتمع الأكاديمي العالمي. يعد كشف الانتحال في الوثائق العربية مهمة صعبة بالتحديد بسبب تعقيد بنية اللغة. تقدم هذه الرسالة نموذجاً وإطارًا للكشف عن الانتحال في المستندات العربية. يستند الإطار إلى تمثيل منطقي للمستند مثل الفقرات ، الجمل ، والكلمات. الهدف الرئيسي من هذا البحث هو تطوير وتنفيذ نموذج الكشف عن الانتحال باللغة العربية "ا د بي د ام" والذي يعتمد على النموذج القادر على كشف الانتحال في الوثائق العربية وآلية البحث عن الوثائق المرشحة المماثلة داخل مجموعة بيانات. من خلال تطوير طريقة ما قبل المعالجة بما في ذلك إزالة الكلمات المستبعدة ، الجذعية والتأصيل. التنفيذ على طريقة تعتمد على المحتوى وتتكون أساسا من بصمات النصوص حسب خصوصية اللغة العربية ومقارنة تمثيلها المنطقي باستخدام خوارزميات الإستدلالية للكشف عن الانتحال. لقد قدمت أداة للكشف عن الانتحال في الوثائق لعربية باستخدام وظيفة تجزئة "ب ك د ر" التجزئة تعتمد عليها لتوليد بصمات النصوص باستخدام دالة الهاش. الهدف الثاني هو تمثيل المستند المنطقي هو توفير وقت الحساب عن طريق تجنب المقارنات غير الضرورية. ولهذا السبب ، قمت بتعريف خوارزمية الإستدلال لكل مستوى في الشجرة: مستوى المستند ومستوى الفقرة ومستوى الجملة. تم اختبار وتقييم نظام "ا د بي د ام" للكشف عن الانتحال في المصادر الإلكترونية للوثائق العربية باستخدام مجموعة من مجاميع البيانات في هذه الدراسة ، حيث أنها تحتوي على ١٠٠وثيقة ، تم جمع ٩٠٪ من المستندات من موقع الويب (كشف انتحال العربية) مقسمًا إلى ثلاث فئات، مجموعة بيانات١ (صغيرة) و مجموعة بيانات٢ (متوسط) ومجموعة بيانات٣ (كبير) ، وتم جمع ١٠٪ من المستندات من وثيقة نظام دعم القرار، تم إنشاء المستندات الأصلية بواسطة الاستبدال عشوائياً مع درجات مختلفة من الانتحال تمت تسميتها مجموعة بيانات٤. في هذه الدراسة ، أجريت التجارب الأولية باستخدام الأداة "ا د بي د ام" و "وي كوبي فايند". حيث كانت النتيجة عند اختبار مجموعة بيانات١ كانت نسبة الإنتحال في "ا د بي د ام" ١٤٪ في زمن مقداره ٥٠١ ثانية في حين "وي كوبي فايند" اكتشفت ٠٪ في ١٣٥ ثانية ، وكذلك اختبرت مجموعة بيانات٢ وكانت نسبة الانتحال ٨٪ في ١٣٧٤ ثانية ، في حين أن "وي كوبي فايند" اكتشفت ٠٪ في ٤٧٥ ثانية. وأيضاً مجموعة بيانات٣ ١٨٪ كشفتها خلال ١٤٣٠ ثانية حيث اكتشفت "وي كوبي فايند" ٦,٣٣٪ في ٢٧١ ثانية ، أما ٩٤٪ مجموعة بيانات٤ فقد سرقت في ١٦٨٢,٧٩ حيث اكتشفت "وي كوبي فايند" ٨١,٤٤٪ في ٣٥٧ ثانية. الاستنتاج الرئيسي هو أن أفضل نتائج "ا د بي د ام" تعاملاً في الكشف الانتحال في الوثائق العربية حين أنها متوسطة في الوقت المستغرق و "وي كوبي فايند". أنها ضعيفة في التعامل مع كشف الانتحال في حين أنها أفضل في الوقت الذي يستغرقه. أظهرت النتائج التجريبية أداء رائع من "ا د بي د ام" حيث حقق قيمة الإستدعاء ٠,٧٨٠٣٥١، بدقة ٠,٩٩٤٢٦٤ ومقياس إف ٠,٨٦٥٦٨٨.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATION

Abbreviation	Meaning
MSA	Modern Standard Arabic
WWW	World Wide Web
BKDR	Brian Kernighan and Dennis Ritchie
LCS	Longest Common Substring
ADPDM	Arabic Documents Plagiarism Detection Model
APA	American Psychological Association
MLA	Modern Language Association
LSA	Latent Semantic Analysis
SVD	Singular Value Decomposition
SCAM	Stanford Copy Analysis Mechanism
IR	Information Retrieval
APD	Arabic Plagiarism Detection
API	Application Programming Interface
UTF-8	Unicode Transformation Format 8-bit
AraPlagDet	Arabic Plagiarism Detection
MDR	Match Detect Reveal
PPChecker	Plagiarism Pattern Checker
SNITCH	Spotting and Neutralizing Internet Theft by Cheaters
HTML	Hyper Text Markup Language
POS	Part of Speech
MBNB	Multi-variant Bernoulli Naïve Bayes
C&P	Copy-and-Paste
TF-IDF	Term Frequency-Inverse Document Frequency
LD	levenshtein distance
APlag	Arabic Plagiarism
FS-APD	Fuzzy Set Arabic Plagiarism Detection
SFS-APD	Semantic-based Fuzzy Set Arabic Plagiarism Detection
AWN	Arabic Word Net
IDE	Integrated Development Environment

JDK	Java Development Kit
JRE	Java Runtime Environment
PHP	Personal Home Pages / Hypertext Preprocessor
HTTP	Hypertext Transfer Protocol
XML	Extensible Markup Language
XAMPP	Cross-Platform (X), Apache (A), MySQL (M), PHP (P) and Perl (P)
TXT	Text File

LIST OF APPENDICES

CHAPTER I

INTRODUCTION

1.1 Introduction

Plagiarism is stealing ideas of others as Nnamani elt define"Plagiarism is the act of taking the writings of another person and passing them off as one's own. The fraudulence is closely related to forgery and piracy-practices generally in violation of copyright laws." [61]

Plagiarism can define as the use of other's work or ideas without proper citation. Detecting and deterring plagiarism strongly needed in many areas. Academic field is at the top of them. According to some studies about academic dishonesty [60], at least 10% of students' work could be plagiarised in USA, Australia and UK universities [63].

Plagiarism became one of all the foremost necessary problems for universities, schools, and researchers. It's really easy through the internet and owing to using advanced program to search out documents or journals by students. A number of the researchers are just repeating and pasting others works without related to the owner of the documents. There are many kinds of plagiarism exist, as well as direct repeating of phrases or passages from a printed text without citing the sources, plagiarism of ideas, sources, and authorship. In addition different kinds of plagiarism, such as translate content to a different language, presenting identical content with different media like pictures, videos and texts, and mistreatment program code deprived of permission. [41]

Arabic language is one of the most important languages which humankind has known over time and ages. It has been knowning since Pre-historic times, and people began to meditate on it. They began to sing their poems, ideas and others. With the beginning of the Islam and its spread on the Arabian Peninsula. With a great deal of interest especially after being associated with the Holy Quran began clear and explicit trends to search for and learn the Arabic language, in order to master the Islamic religion, and to identify its concepts ,ideas, and manifested the Arabic language, it became one of the most important languages. It is becomes well known, as mother tounge. it uses for poetry, proverbs .prose and rhetoric .It is based on the principle of learning in the first and last postion. It is no wonder when we classify as the most

important and famous in the history of humanity as a whole. Arabic belongs to the Semitic language group. The main characteristics of Modern Standard Arabic (MSA) [72, 73] , It is written from the right to the left. Its basic alphabet and contains 28 letters, one of these letters are 3 long vowels and eight other forms: The hamza with six forms, the ta marbouta and the alif maksour as well as the ligation of the letters (L) and, which is written (called lamalif). - A special feature of the Arabic language is that the letters change shape depending on their location in the word. They are many Homographs are disambiguated using the diacritics in Arabic language ([41], [2]) and [19] [1].

1.2 Research Problem Background

Most work in document plagiarism has been prepared for academic purpose. Detecting plagiarism is important to judge and to identify students' work, especially for postgraduates who strictly not allowed for cheating, rewording, rephrasing, or restating without references. Regarding, numerous plagiarism detection systems has been developed for Arabic documents. Most of these systems use plagiarism techniques known as similarity detection techniques, which create special "fingerprints" for collecting files, including metrics, such as average line length, file size, average number of commas per line. Clearly, small fingerprint records can be compared rapidly, but this technique is now considered unreliable and rarely used nowadays [3].

1.3 Research Problem Statement

Huge information of Arabic language are available on the World Wide Web (www) and digital libraries, so it is very difficult to find an Arabic passages from different source. [41][78]. Then, it is a research challenge to universities, schools and researchers especially when putting on consideration the extreme verbatim and complexity in Arabic language. In recent years, there have been several types of ways to search and detect plagiarism although those regarding the text in the Arabic language have been very restricted [41][78]. Due to the lack of an extensive study on plagiarism widespread in the Arab world, researchers are suffering from this problem as well as attention of a huge total news is certify the reasaerches on this topic. There are many studies in

plagiarism among Arab education revealing some insufficient awareness about the attraction and description in plagiarism.[41][78]

1.4 Research Objectives

The objectives of this research are summarized as follows:

- To introduce Arabic documents plagiarism model.
- To develop tools of Arabic plagiarism detection based on introduced model and framework which capable on detecting plagiarism in Arabic documents and search mechanism for the similar candidate documents within the corpus collection by developing pre-processing method including stop-word removal, stemming and rooting.
- To evaluate the effectiveness of provides Arabic plagiarism detection tools.

1.5 Research Scope

In order to achieve the objective stated above, the scope of this research is focus on detection of plagiarism in Arabic documents (only Arabic).

1.6 Research Methodology

The main aim of this research is to develop and implement the proposed Arabic documents plagiarism detection model "ADPDM" tools. Which are already has mentioned on the research objectives. In our implementation which built around a content-based method consisting mainly in fingerprinting the texts according to Arabic language specificity and comparing their logical representations by using heuristic algorithm we introduced a plagiarism detection tool for Arabic language by using the BKDR (comes from Brian Kernighan and Dennis Ritchie) [22] Hash function for chunk(3-gram) hashing. This function returns the sum of multiplications of each character by a special value (named seed and usually equal to 31); Seed value should be a prime number. The second aim of the logical document representation is to save

computation time by avoiding unnecessary comparisons. For this reason, we define a heuristic algorithm for each level in the tree: document level, paragraph level, and sentence level. We measure it using the Longest Common Substring (LCS) metric.

1.7 Thesis Organization

This thesis has organized into Seven Chapters as following: Chapter has deal with research introduction, problem background, problem statement and objectives etc. Chapter II shows the Literature Review. In Chapter III, the researcher has described the Arabic documents plagiarism detection model. Chapter IV deals with framework of plagiarism detection framework and tool. Chapter V shows the development plagiarism detection Arabic documents. Chapter VI explains experimental work done and Dissection, the last Chapter employed with summary and future work.

CHAPTER II

LITERATURE REVIEW

2. Introduction

This chapter introduces a definition of plagiarism, type of plagiarism and literature review of plagiarism in Arabic documents.

2.1 The Plagiarism

Plagiarism is defined as the unauthorized use or close imitation of the language and thought of authors and their representation as one's own original work [26][1]. It involves literary theft, stealing (by coping) the words or ideas of someone else and passing them off as one's own without recognizing the source. Many people think of plagiarism as copying another's work, or borrowing someone else's original ideas. However, terms like "copying" and "borrowing" can disguise the seriousness of the offense [27][1].

Plagiarism detection is a sensitive field of research, which has gained lot of interest in the past few years. Although plagiarism detection systems are develope to check text in a variety of languages, they perform better, when they are dedicated to check a specific language as they take into account the specificity of the language, which leads to better quality results [64].

Plagiarism becomes one of the most important issues for universities, schools, and researchers [20]. It is so easy through the internet and due to using advanced search engine to find documents or journals by students. Some of the researchers are just copying and pasting others works without reference to the owner of the documents. Several types of plagiarism exist, including direct copying of phrases or passages from a published text without citing the sources, plagiarism of ideas, sources, and authorship. There are other types of plagiarism, such as translating content to another language, presenting the same content with other media like images, videos and texts, and using program code without permission. [2]

According to the Merriam-Webster Online Dictionary ("Plagiarism", 2007), to"plagiarize" means: To steal and pass off (the ideas or words of another) as one's own, to use another's production without crediting the source, to commit literary theft. Alternatively To present as new and original an idea or product derived from an existing source.

Plagiarized document detection plays important roles in many applications, such as file management, copyright protection, and plagiarism prevention. [27,1]. Plagiarism can take one of the popular types such as copying of the whole or some parts of the document, rewording the same content in different words, using others' ideas or referencing the work to incorrect or non-existing sources [9,1]. Other ways of plagiarism include translated plagiarism wherein the content translated and used without referencing the original work, artistic plagiarism in which different media such as images and videos are use to present other's work without proper citation [15].

Citating to avoid plagiarism is one of the effective ways is Citation. Follow the document formatting guidelines used by your educational institution or the institution that issued the research request. This usually, entails the addition of the author(s) and the date of the publication or similar information. The citation provides a summary description of the book, article, web page, etc. Includes the author, title, name of periodical and volume, publisher, date, and alternative characteristic data.Is a manner reference you tell your readers that sure material in your work came from another supply ,it gives a concise description of the book, article, web page, etc.Includes the author, title, name of periodical and volume, publisher, date, and other identifying information. [60]

Quoting A quotation is the repetition of one expression as part of another expression, especially when the quoted expression is clearly known or explicitly attributed by quotation to its original source and is indicated by quotation marks. [60]

A **reference** collects the identification data of the specific source we cited or paraphrased. Every reference may go immediately following the citation or paraphrased quote between brackets (…reference…) and/or be part of a numbered list of references [1] [2] [3]… at the end of our document or at the end of a section or page.

Paraphrasing is presenting the ideas and information you have read in your own words - is an important academic skill. By translating content from your research into your own words, you demonstrate to your reader that have understood and are able to convey this content [60].

Summarization is an overview of content that gives a reader with the overarching theme. Summaries will save a reader time because it prevents the reader from having to really bear and filter the vital info from the unimportant [60].

2.2 Types of Plagiarism

There are different types of plagiarism and all are serious violations of academic honesty. There may be cultural differences in the definition of plagiarism. The main type of plagiarism can divide in the following [38]:

2.2.1 Intentional Plagiarism

Intentional Plagiarism is claiming sole authorship of a work that you know to have been largeling written by someone else. It is happens when you claim to be the author of work that you know was originally written completely or in part by someone else, as showing in figure 2.1 the type of plagiarisms[33].

- **Word Plagiarism or Copy & Paste** The plagiarist finds a useful source and copies a portion of that, perhaps with a few minor changes, into the text that is to be changing the name of the author [34]. It is kind of plagiarism that is quickly recognizable and generally granted on to be plagiarism [33].

- **Structure Plagiarism** this sort of plagiarism is troublesome to regulate, mutually should scan each texts terribly closely to envision what has been taken, other when you paraphrase poorly, and even with citation it may be considered plagiarism. [32][33]

- **Style Plagiarism** is follow source material sentence-by-sentence or paragraph-for-paragraph. Although none of your writing does not exactly match the source material, but what is the thinking here, copy it someone else's style. [32]

- **Idea Plagiarism.** Any time you present an idea that's not your own, you must properly cite and reference the source. This can get tricky because sometimes you might think your idea is truly your own original idea. The research paper authors have a hard time distinguishing the ideas and/or solutions provided by the author of the source paper from public domain information. [32][33]. Public domain information is any idea or solution about which people in the field accept as general knowledge [6].

- **Metaphor Plagiarism.** "Metaphors are used either to make an idea clearer or give the reader an analogy that touches the senses or emotions better than a plain description of the object or process. Metaphors, then, are an important part of an author's creative style" [4][37]. to use the same metaphor as another writer, you need to properly cite it.

- **Author Plagiarism.** Here the author of the research paper reuses his own previous work to produce a new work [7].

- **Self-Plagiarism.** is the use of one's own previous work in another context without citing that it was used previously .This type of plagiarism may be new to you, but it's one you need to be aware of.[40]

- **Mosaic plagiarism** Patchwork paraphrasing refers to getting content from various sources line to constant topic of interest and revising the sentences, shift words, exploitation synonyms and improvising on the grammar designs to finally manufacturing one's own analysis paper while not citing the sources [31][33]

- **Shake & Paste** In this type, taking paragraphs from a number of different sources is known without a functional order [32][33].

- **Disguised Plagiarism**. Copy text from source then some effort is made in order to hide the release. You can remove or add words, change the order of words, or even try to redraft. However, the source is not given, or given only to part of the text taken, this is still considered a literary theft [32][33].

- **Plagiarism by Translation.** Plagiarism through translation is taking text from one language and translated either manually or automatic translation assistance system, and then used without naming the source[32][33].

2.2.2 Unintentional Plagiarism

Also referred to as accidental plagiarism this refers to an instance in which it appears that a part of work has been plagiarized when in fact the person who wrote the piece of work did not intentionally set out to commit an infraction [32][33]. As showing in figure 2.1 the type of plagiarisms.

- **Poor Paraphrasing.** change a few words while still keeping the overall sentence structure, or switching the sentenced structure around but not changing any words, it can easily look like youve committed plagiarism.[32][33]

- **Poor Quoting.** That takes a misplaced quotation mark getting a few of the words wrong in a quotation and it might make someone think you've committed plagiarism. To avoid poor quoting must make sure you double and triple-check your quotations to ensure that they are completely accurate and hone to perfection your paraphrasing and quoting techniques .[32][33]

- **Poor Citation.** Forgetting a citation here and there definitely looks like plagiarism to anyone checking or grading your work.[33,32].

2.3 Plagiarism Detection Techniques

Plagiarism detection techniques are known as similarity detection techniques [27]. Latent Semantic Analysis (LSA) [5] is a technique used to describe relationships between a set of documents and terms they contain. In this technique, words that are close in meaning are assumed to occur close together. A matrix is constructed in which rows represent words, and columns represent documents. Every document contains only a subset of all words. Singular Value Decomposition (SVD), a factorization method of real or complex matrix, is used to reduce the number of columns while preserving the similarity structure among rows. This decomposition is time consuming because of the sparseness of the matrix. Words are compared by taking the cosine of the angle between the two vectors formed by any two rows. Values close to 1 represent very similar words, while values close to 0 represent very dissimilar words this technique is suitable for Arabic plagiarism detection. Stanford Copy Analysis Mechanism (SCAM) [7] is based on a registration copy detection scheme. Documents are registered in a repository and then compared with the pre-registered documents. The architecture of the copy detection server consists of a repository and a chunker. The chunking of a document breaks up a document into sentences, words or overlapping sentences.

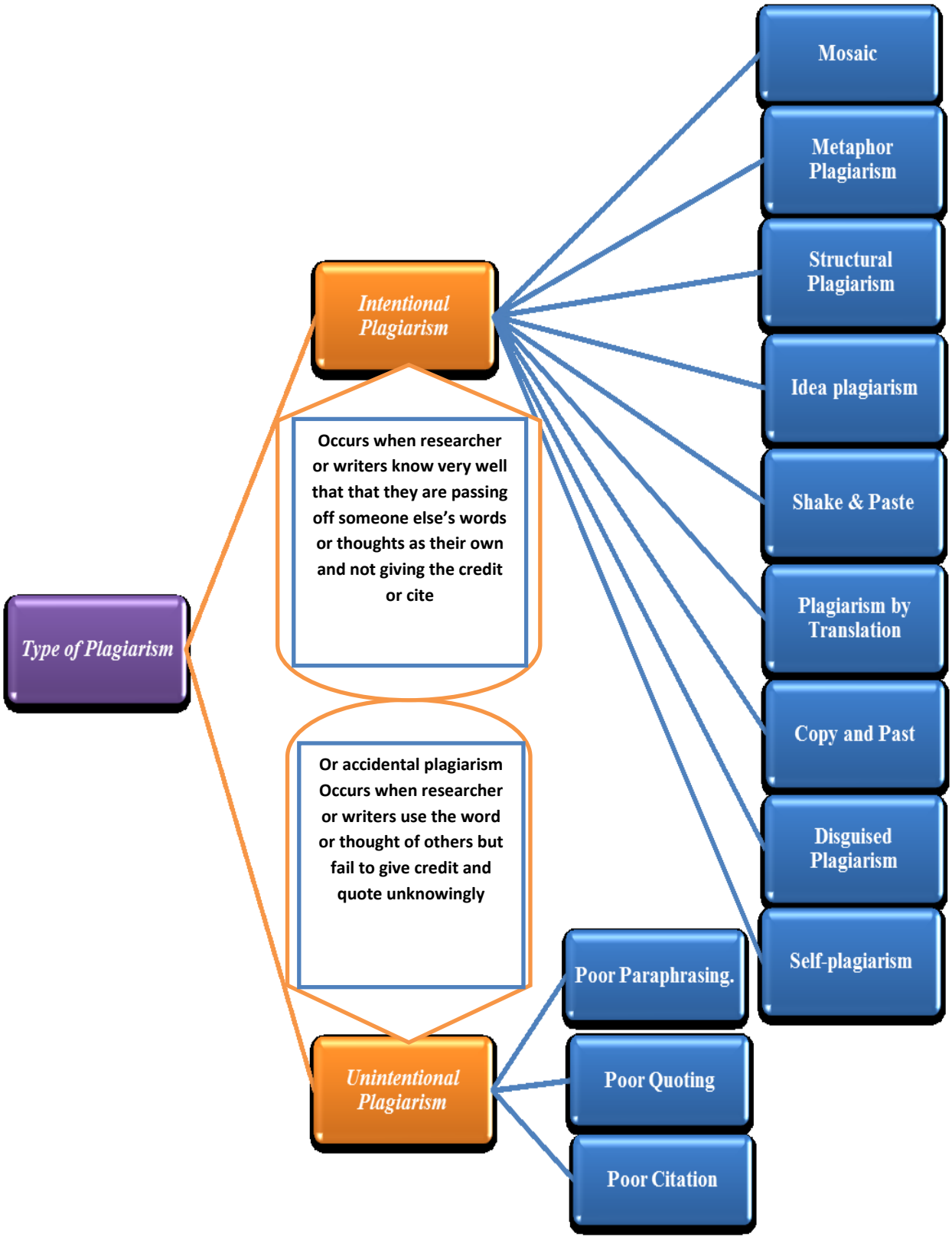


Figure 2.1 Type of Plagiarisms [32]

The most popular techniques include string tiling, finding the joint coverage for a pair of files [19, 20] and parse tree comparison [21,22]. Usually these techniques work in pairs of files, so the comparison routine should called for each possible file pair found in the input collection.

Alzahrani and Salim present statement-based plagiarism detection technique in Arabic scripts using fuzzy-set IR model in which the degree of similarity is calculated and compared to a threshold value to judge whether two statements are the same or different. They construct and test documents with about 250 plagiarized statements, their results show that fuzzy set IR successfully detected not only exact but also similar statements that have different structure [23,24].

A fingerprint is a set of integers created by hashing subsets of a document to represent its key content. Techniques to generate fingerprints mainly are base on k-grams (a k-gram is a contiguous substring of length k) which serve as a basis for most fingerprint methods [17]. Fingerprinting technique is widely used for Arabic plagiarism detection. K-grams are central to fingerprinting techniques because fingerprinting divides the document into grams of certain length k [24]. This allows the fingerprints of two documents to compare in order to detect plagiarism. The fingerprint matching approach differs based on the comparison unit (i.e., grams)[12].

2.4 Way and strategy to Avoid Plagiarism

It is easy to find information for most research papers, but it is not always easy to add that information into your paper without falling into the plagiarism trap. There are easy ways to avoid plagiarism. Follow some simple steps while writing your research paper to ensure that your document will be free of plagiarism.[65]

2.4.1 Specific words and phrases

Use author's specific word or words, you must place those words within quotation marks and you must credit the source.[65]

2.4.2 Information and Ideas

Information: The information depends on part of the common knowledge you will need to provide a source of and then document it. Ideas: The author's ideas may include points reached, conclusions drawn, his method of a specific theory, or a list of steps in a process or characteristics.[66]

2.4.3 Common Knowledge

General common knowledge is information considered to be in the public domain, such as birth and death dates of well-known figures, and generally accepted dates of military, political, literary, and other historical events. In general, information contained in multiple standard reference works usually is considers in the public domain [66]. In addition, in the case of both general and field-specific common knowledge, if you use the exact words of the reference source, you must use quotation marks and credit the source. Field-specific common knowledge is "common" only within a particular field.[66]

To avoid plagiarism they are eight guides as following:

Firstly give credit where credit is due when paraphrasing. Always use your own words when using someone’s ideas, information, or analysis. Remember to use all original language when paraphrasing a source. You need to use your own style and your own words when paraphrasing! Both stealing words and/or style is plagiarism. [67][68]

Secondly you have to Give credit where credit is due when directly quoting. When quoting a sentence, put the person's words in quotation marks and include an APA formatted in-text citation. [67][68]

Third Citing a quote can be different from citing paraphrased material. This practice usually involves the addition of a page number, or a paragraph number in the case of web content.[67][68]

Fourth, reference page or page of works cited at the end of your research paper. Again, this page must meet the document formatting guidelines used by your educational institution. The author(s), date of publication, title, and source is is very specific information. Follow the directions for this page carefully. You will want to get the references right [67][68].

Fifth, Add your own analysis or thoughts after you have inserted directly quoted words or paraphrased knowledge. This allows you to put your own spin on the research you have used. It also allows you to illustrate the explicit connection between the research you chose and your essay's intent or thesis statement. [67][68]

Sixth, use a plagiarism checker to see if you plagiarized. Keep your similarity index below 15%. "In research papers, you should quote from a source to show that an authority supports your pointand to present a particularly well-stated passage whose meaning would be lost or changed paraphrased or summarized"[67][68].

Seventh, the plagiarism checker, marking your work as suspect, will likely flag Reused work. You can reference former papers you wrote or have published, but you cannot present your previously written work as new. To do so, is academically dishonest [67] [68].

Eight avoid copy from the web this will be easily flagged by the impersonation checker or by inserting suspicious text into Google [67][68].

2.5 Characteristics of Arabic Language

Arabic language is the language belongs to the Afro-Asian cluster, which contains a many of privacy, making it completely different from various Indo-European languages. It has 28 letters of the alphabet letters (أ, ب, ت, ث ... ي) ,Three of them are long vowels letters like (ا, و, ي) besides those residual are consonants letter as showing in table 2.1 **[41]**. Arabic letters modification per their position within the word, and should be elongated by using a special dash between 2 letters[19]. The direction for writing Arabic is right to left, cursive, and doesn't contain capitalization. Discretization the Arabic is to feature and symbol (diacritic) on top of or below the letters to point the right pronunciation and which give the meaning of the word. In the absence of individualization most Arab media both electronic and print a challenge to understanding the Arabic language. Arabic language can be a pro-drop whish permits subject pronouns to drop, like in Italian, Spanish, and Chinese [19]. There are diacritics (العلامات الإعرابية) which are (_َ _ُ _ِ _ْ"). 1) "َ"The fathea character appearing on top of a letter to give the "a" sound. 2) "ُ" the Dahamma character appears on top of a letter to give the "u" sound. 3) "ِ" the kaasra character appears below a letter to give the "i" sound, and 4) "ْ" the soukuun character showing on above of a letter to point that no sound from the previous ones to thereto letter. They are many Homographs are disambiguated using the diacritics in Arabic language.

However, Arabic letters differ in shape depending on whether the letter comes in the beginning, middle or end of the word. it has many different local dialects. Yet the Arabs can understand nearby dialects easily, and some of the other dialects. Although, they can communicate easily if they use the Standard Arabic language.

The Arabic word from the stem may consist of affixes and "including some prepositions, conjunctions, determiners, and pronouns". It obtained by adding affixes to stems that are successively obtained, by adding affixes to the roots. As an example, the word المساجد, translated Al-masajid and meaning mosques, which is derivative from the

stem مسجد, translated masjid, meaning Mosque, which is derivative from the root سجد, transliterated sajid, and meaning to write [19,41].

Table 2.1: The Arabic Alphabet Vowels

Name	Character	Explanation
Damma	◌ُ	Damma is an apostrophe-like shape written above the consonant which precedes it in pronunciation. It represents a short vowel u (like the "u" in "but").
Fatha	◌َ	Fatha is a diagonal stroke written above the consonant which precedes it in pronunciation. It represents a short vowel a (a little like the "u" in "but"; a short "ah" sound).
Kasra	◌ِ	Kasra is a diagonal stroke written below the consonant which precedes it in pronunciation. It represents a short vowel i (like the "i" in English "pit").
Sukūn	◌ْ	Whenever a consonant does not have a vowel, it receives a mark called a sukūn, a small circle which represents the end of a closed syllable . It sits above the letter which is not followed by a vowel.
Shadda (or tashdīd)	◌ّ	Shadda represents doubling (or gemination) of a consonant. Where the same consonant occurs twice in a word, with no vowel between, instead of using consonant + sukūn + consonant, the consonant is written only once, and shadda is written above it.
Alif	ا	Alif is the long vowel ā (a long "ahh" sound as in English "father").
Wāw	و	Wāw is the long vowel ū (like the "oo" in "moon"). It also represents the consonant w. When Waw is used to represent the long vowel, damma appears above the preceding consonant.

Ya'	**ي**	Ya' is the long vowel ī (like the "ee" in English "sheep"). It also represents the consonant y. When Ya' is used to represent the long vowel, kasra appears above the preceding consonant.

Many languages-sensitive tools for detecting plagiarism in natural language documents have been developed, particularly in English. It is also exist, but it is restrictive because it usually does not take into account the specific language features. Most of the issues of plagiarism have occurred for a protracted time, however with the advances in data technology, and drawback worse[11]. There are many tools, which have been used to detect the plagiarism. These tools were developed only to detect English version, while other tools were adapted to deal with French, German and Chinese languages. However, for the Arabic language, these tools are under development and no commercial products are available yet tell now. Therefore, this research is amid to design tool for plagiarism detection Arabic documents, to facilitate the process of plagiarism detection, trace and estimate the degree of plagiarism in any Arabic text document. [11]

2.6 Plagiarism in Arabic Documents:

Despite the lack of large-scale studies of the widespread plagiarism in the Arab world, this problem had attention from the large number of news which attest its pervasiveness. There are also some studies that show the lack of awareness on the definition and seriousness of plagiarism among Arab educative[16,49,78].

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 for Arabic documents[15]. Now it is applied to all educational levels both in secondary and university level. Most of these systems use plagiarism

techniques known as similarity detection techniques, which create special “fingerprints” for collecting 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 [3]. Ameera Jadalla and Ashraf Elnagar in (2012) proposed Iqtebas 1.0, which is a primary solid and complete piece of work for plagiarism detection in Arabic text files. It is similar to a search engine. The goal of the Iqtbas 1.0 is to compute the originality, value of the examined document, by computing the distance between each sentence in the text and the closest sentence in the suspected files [2]. Farahat F. Farahat, et al in (2015) are tested experimentally ZPLAG. This is prototype for detecting plagiarism in documents written in Arabic language, where some hidden plagiarism forms can be detected, such as change of sentence structure and replacement of synonym. 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[27].

2.7 Fingerprint Matching Technique

Fingerprinting techniques mostly rely on the use of K- grams (Manuel et al. 2006) because the process of fingerprinting divides the document into grams of certain length k. Then, the fingerprints of two documents can compare in order to detect plagiarism. It has been observes through the literature that fingerprints matching approach differs based on what representation or comparison unit (i.e.grams) is used.

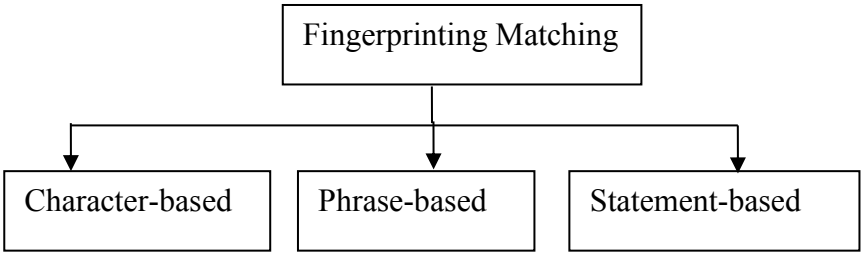


Figure 2.2: Fingerprint Matching Technique

2.7.1 Character-based Fingerprint Matching

The conventional fingerprinting technique uses sequence of characters to form the fingerprint for the whole document. During 1996, Heintze divides fingerprinting techniques into two types, which are full and selective. Infull fingerprinting, document fingerprint consists of the set of all possible substrings of length K. For example, if we have a document of length |D| = 5 consisting only one statement that has only one word "touch", then we can see that "touc" and "ouch" are the all possible substrings of length K = 4. In general, there are |D| – k + 1 substrings or k-grams, where |D| is the length of the document. Comparing two documents under this technique is counting the number of substrings that are common in both fingerprints [75].

2.7.2 Phrase-based Fingerprint Matching

In 2001, Lyon et al. generates fingerprint using phrase mechanism to measure the resemblance between two documents. During the early stage, we have to convert each document to a set of trigrams (three words). Hence, a sentence such as "*Web Based Cross Language Plagiarism Detection*" will be converted to the set trigrams {"*Web Based Cross*", "*Based Cross Language*", "*Cross Language Plagiarism*", "*Language Plagiarism Detection*"}. Then, the set of trigrams for each document is compared with all other using the matching algorithm. Finally, the measure of the resemblance for each pair of documents is calculated.[18]

2.7.3 Statement-based Fingerprint Matching

The pros and cons of character-based and phrase-based fingerprinting have led Yerra and Ng (2005) to represent the fingerprints of each statement (and thence the whole document) by three least-frequent 4-grams. Although any value of K can be considered, yet K = 4 was stated as an ideal choice by Yerra and Ng (2005). This is because smaller values of K (i.e., K = 1, 2, or 3), do not provide good discrimination between sentences. On the other hand, the larger the values of K (i.e., K = 5, 6, 7...etc), the better discrimination of words in one sentence from words in another. However each K-gram requires K bytes of storage and hence space consuming becomes too large for larger values of K. Therefore, we can conclude that K = 4 is an optimal or near optimal

choice. Here is an explanation of how this 3-least frequent 4- grams works.A 4-gram of a string is a set of all possible 4-character substrings. For example, let take a string S = "English Word", then the possible set of 4-grams include"*engl,ngli, glis, lish, ishw, shwo, hwor, word*" with ignoring spaces.[79]

Secondly, three least-frequent 4-grams are the best option to represent the sentence uniquely. To illustrate the three least-frequent 4-gram construction process, consider the following sentence S "soccer game is fantastic". The 4-grams are socc, occe, ccer, cerg, etc. In this method, instead of comparing all possible 4-grams, only three 4-grams that have the least frequency over all 4-grams will be chosen. [3].

2.8 Plagiarism Algorithms

As we can notice of the plagiarism, there are several methods to detect plagiarism; we have a tendency to differentiate between two kinds of methods that to find out plagiarism (language independent methods and language-sensitive). Base on an independent method, the assessment the characteristic of the text, this is not inherent in particular natural language, like the number of single figures, the median sentence extent, called Language-independent method [35]. The language-sensitive is bases on a sensitive way to evaluate the text attributes that are specific to one language [35].

Further methods impressed by authorship attribution, referred to as stylometry-based methods, and may be utilized in language sensitive systems. We provide a number of the main points of those methods within the following subsections.

Content-based methods

Base on Consisting of text analysis specifications in terms of logical structure to detect similarities. Furthermore, it is has place confidence in specific comparisons of the

document contents during an exact illustration. All these method they deal with stopping "deleting stop-words" and rooting "decreasing words to practical root formula" procedures. Tools to detect plagiarism revolve around Content-based methods, contain CHECK [57], Wcopyfind [25,56], Turnitin [28], and EVE2 [58]. Additionally, in and advancing practices of hidden plagiarism transform words to their greatest popular synonyms can be help to detected.

Fingerprinting is one of the greatest common techniques used for plagiarism detection [44]. It changes the content of the document to a collection of integers [7]. The produces are integers by the hashing divisions of a document. Fingerprints will measure their similarity. There and a lot of there are many techniques to produces fingerprints. The foremost acknowledge one is predicates on k-grams - A k-gram could be a string of length k from the document. There are several ways wont to choose fingerprints, like choosing each i hash of the document, and therefore the winnowing technique supported windows containing hours[8].

As noted, a Technique utilized in language process to explain the connections between a set of terms and documents are named Latent Semantic Analysis (LSA). LSA is principally supported a matrix among that rows are the terms and columns in the documents [5,17].

SCAM is stands for Stanford Copy Analysis Mechanism which based on registration-copy-detection scheme. A pre-registered repository is maintained and any new document is compared against this repository. This repository and a "chunker" are part of the copy detection server. The document is chunked before being registered. The chunker breaks the document into smaller units as sentences, words or overlapping sentences. The new chunked document compared unit by unit with the repository/pre-registered documents. Inverted index storage is used for sorting chunks of registered documents. The units contained within the document is a pointer to the document within which the chunk exits i.e. posting. Each posting is segmented. First segment is the

“document name” and the second segment is the related chunk occurrence number. The small unit of chunk raises the similarity level between documents. Each chunk unit in SCAM is “word”. The comparison to the repository of the document is performed using Relative Frequency Model, i.e computing the frequency of group of words among two documents [17,11].

A method of retrieving the information has been use to found out a match between the query and documents are called ranking. It uses similarity measure to calculate the scores of games and query documents are sorted fade from their findings. Highly ranked document are then returned [42].

Hoad and Zobel suggested several different formulas to measure the similarity supported the quantity of events of comparable words in documents, like the length of the document, the difference of the frequency of the word in the query and documents, and a weight measurement of the weight of importance term [42].

APD is stand for Arabic Plagiarism Detection tool dedicated to the Arabic language [14]. Which is based on the fingerprint of each document submitted by taking 4 grams less frequent and compares them to a group within the Corpus of fingerprints document. It is then used in the formation of recovery technique based on fuzzy sets to detect matches between documents.

Stylometry-based methods

Stylometry is a statistical approach used for authorship attribution. It is based on the assumption that every author has a unique style [35]. Writing style will be analyzed using the factors inside constant document, or by comparison the 2 documents from the author himself The supposed plagiarism detection inside constant document and while not considering external references, plagiarism detection considerably [31].

Stylometry-based methods can be used in internal and external detection, but content-based methods can be used only in external detection. Moreover, if an author has more than one style, stylometry-based methods can detect false-positive plagiarism. Content-based methods are generally better than stylometry-based methods in terms of precision [16] and can give a proof of plagiarism by visualizing the results.

We distinguish among the plagiarism detection tools, “Stylometry-based” and those called “Content-based”, the former being more oriented towards the intrinsic plagiarism detection while the latter is designed for detection of external plagiarism. Detecting external plagiarism is, according to [23], “about searching for sources of a suspicious document” whereas the intrinsic detection, according to the same source, is “about identifying plagiarized passages via Breaches of writing style”. Research in the field of plagiarism detection in Arabic, or at least those known to us, are almost all “Content-based”. The approach adopted is substantially the same in a large number of researchers [17, 18, 19, 20, 21, 22], at least in that it includes two steps:

- A first step of pre-processing, consisting of a tokenization of the text, the so-called stop-words removing, then the rooting.

- This second step, when it comes to “Content-based” research, is to study the values returned by a hash-function (Fingerprint), the degree of similarity between documents based on the Fuzzy IR (Fuzzy Set Information Retrieval) model, or to group documents into clusters based on their degree of similarity (Clustering).

Turnitin.com is used to match the digital papers presented against online resources and a database in the former house of the papers submitted fingerprints. All

papers are archived on auditing in the future - a feature that is especially useful if the suspected copies of former student's papers. [31]

The plagiarism prevention methods that include punishment measures, procedures, and interpretation of plagiarism drawback ways, and plagiarism detection methods that involve manual methods with software tools [2], these Are two main classes of methods used to ease plagiarism. These methods have a semi-permanent positive result, however it needs an extended time to implement, meanwhile they have confidence on social cooperation between very different universities and departments to decrease plagiarism [1], each method could be combined to reduce deception and cheating. However, the software package tools are the best way for verify plagiarism, and may be the ultimate arbiter manually [3].

Winnowing algorithm: The winnowing algorithm is an algorithm to select document fingerprints from hashes of k-grams [8]. To obtain the fingerprint of a document, the text is divided into k-grams, the hash value of each k-gram is calculated, and a subset of these values is selected to be the fingerprint of the document[8].

Meni in 2012 introduce APlag, a new plagiarism detection tool for Arabic texts, based on a logical representation of a document as paragraphs, sentences, and words, and new heuristics for text comparison. We describe its main attributes and present the results of some experiments conducted on a dummy test set. We demonstrate its effectiveness by comparing its performance to that of APD, a plagiarism detection tool for Arabic. Overall, preliminary results show that APlag significantly improves the results obtained by APD in terms of recall and precision metrics[19,11,41].He implementation of a prototype of APlag in Java and evaluate their performance on a hand-made test data set of 300 Arab and close to about 800 words each. We extracted 20 documents of different books available on the site Alwaraq [11]. He was generated three data sets of original documents as follows: Data sets synonymous and used to change the

structure to evaluate the performance APlag to detect plagiarism hidden. Data set all the data served to measure the performance of APlag above all to detect plagiarism hidden an exact copy of parts of the texts.[41]

Kamal [21] has developed APD Tool stand-alone desktop tool base on Winnowing local document Fingerprinting Algorithm.it has been adaptive for Arabic and tested using three essays written by a class of Student. She has concluded that ADP is an efficient solution to minimizing student coping.

"Bing" is a search engine, they developed a system to detect plagiarism in both Arabic and English languages. The system which relies on plagiarism detection algorithm is effective and can support both Arabic and English languages. Through experiment and tests on our plagiarism detection algorithm, we found that this algorithm reduced the un-useful comparison between texts, since it compares only between cue-phrases surrounding words which forms the logical and natural boundaries of text sentences [13].

Alzahrani et al., 2009 have produced an Arabic plagiarized detection (APD) tool especially for working with Arabic language [30,45]. APD 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 [30].

PlagScan supports all the language that use the international UTF-8 encoding and all language with Latin or Arabic characters can be checked for plagiarism Supported Languages: CheckForPlagiarism.net supports English languages, Spanish, German, Portuguese, French, Italian, Arabic, Korean, and Chinese languages [47]. And iThenticate supports more than 30 languages, it mean that it supports most of languages likes ”English, Arabic, Chinese, Japanese, Thai, Korean, Catalan, Croatian, Czech, Danish, Dutch, Finnish, French, German, Hungarian, Italian, Norwegian, Polish, Portuguese, Romanian, Serbian, Slovak, Slovenian, Spanish, Swedish, Greek, Hebrew, Farsi, Russian, and Turkish.” [47].

“AraPlagDet”[Arabic Plagiarism Detection] is the first shared task that addresses the plagiarism detection in Arabic texts in “PAN plagiarism detection competition”[31].

Many researchers adopted this idea for their knowledge development and raising of the awareness level on the plagiarism problems and the importance of its detection in the Arab world. Modern plagiarism detection systems usually implemented using certain content-comparison techniques. The most popular techniques include string tiling, finding the joint coverage for a pair of files [13, 46] and parse trees comparison [15, 49 ,17]. Some of existing plagiarism detectors that employ structure-based methods such as plagues (one of the earliest structure-based detectors). [43]

Other approaches have been used for plagiarism detection which includes “Swarm Summarization” [69] of documents. The idea is to use a summary of the suspected document as query to send to a search engine and [69] conducted even to a “dictionary-based translation” to bring documents from the web in foreign languages. In

another approach, briefly described in a short paper [70] proposes to rely on a text mining tool. The benefit would be a reduction of pre-processing, the “tokens” being extracted by the text mining tool and stored in an archive. A specific text mining tool is proposed, in this case the open source software RapidMiner [71]. This tool offering no option for processing Arabic documents, the authors plan to develop an “add-on” for it[64].

2.9 Plagiarism Detection Tools for Natural Language Documents

Several tools have been developed for plagiarism detection. They use variety of document descriptors that entail different techniques. Here is a brief exploration of eleven plagiarism detection tools: Diff, SCAM, SIF, COPS, KOALA, CHECK, MDR, PPChecker, SNITCH, WCopyFind, and Ferret. They are also summarized in Table 2.5.

Diff is a Unix/Linux Command (Yerra and Ng, 2005) that uses line-based representation for source code, text, and other line-oriented files. It compares files line-by-line and captures the differences between two text documents one line at a time.

SIF, developed by Manber (1994), finds similar documents by using the fingerprinting scheme to characterize documents. However, it cannot measure the degree of overlap between two documents nor display the location of plagiarism. Moreover, if files containing the same information but using different sentence structures, they will be considered dissimilar.

SCAM (Stanford Copy Analysis Mechanism), developed by Shivakumar (1995), performs word-based copy detection, does not specify the plagiarism locationand can handle only small documents.

COPS, developed by Brin (1995), uses hash-based scheme for copy detection. It compares hash values of given documents with that in the database for copy detection.

COPS has several limitations reported by Yerra and Ng (2005). First, the use of hash function produces large number of collisions. Next, documents to be compared by COPS must have at least 10 sentences. Lastly, it has problems selecting correct sentence boundaries.

KOALA, designed by Heintze (1996), selects substrings of a document based on their usage and compares their fingerprints. This results increase the accuracy of KOALA in comparison to COPS.

CHECK is a structured-based plagiarism detection system developed by Antonio et al. (1997). It has some mechanism to determine the subject related to the document and then search domain is limited to only document with the same or relevant subjects. CHECK studies the semantics of the documents in addition to their syntax and is applied to only documents discuss same subject until two paragraphs which are highly related semantically are found. The paragraphs are then compared in detail, i.e., on a sentence-per-sentence basis, to determine plagiarised paragraphs.

MDR (Match Detect Reveal) system was developed by Zaslavsky et al. (2001) to detect plagiarism in documents. It uses suffix-tree representation to index the documents in a digital library. **MDR** applies string-matching algorithms based on suffix trees to identify the overlap between a suspicious document and candidate documents. It is very powerful for finding exact copy. However, constructing suffix tree for documents is very expensive. Besides, this system is very weak at detecting modified documents.

PPChecker (Plagiarism Pattern Checker in Document Copy Detection) was developed by Kang et al. (2006). It uses statement-based representation for original documents and query document. The degree of similarity between two statements is calculated using "local-similarity-extractor" function proposed by the author. Then,

"document-similarity-extractor" function is used to find the degree of overlap between two documents.

SNITCH (Spotting and Neutralizing Internet Theft by CHeaters) was developed by Sebastian and Thomas (2006) to detect copy and paste (exact match) plagiarism in paragraph-based representation. SNITCH implements a fast and accurate plagiarism detection algorithm using the Google Web API. It uses a sliding window to scan documents and locate candidate passages that might be plagiarised. The sliding-window mechanism works as follows. First, SNITCH reads a window containing certain number of words. Then, it calculates the number of characters in each word. After that, the weight of the window is measured as the average of the number of characters per word and the words in the window. Next, the program stores the window's weight for use later. The process will be repeated for all such windows in the document by shifting the window forward in the document one word at a time. SNITCH, then, orders windows in decreasing order according to their weights, eliminates overlapping windows, and selects the top N weighted windows. Lastly, it searches the Internet for each, gathering the top search result (if any) for each. The output is an annotated HTML report containing the original document with hypertext links inserted for any passages that were found on the Internet.

WCopyFind developed by The University of Virginia (2006). It uses phrasebased representation with six or more words as a unit of comparing. It counts thenumber of words from matching phrases and calculates plagiarism rate as a ratio of the number of matching words and the total number of words in the document.WCopyfind could find a partial overlap, but the user should set an adequate word number in a phrase.

Ferret (Lyon et al. 2001; Lyon et al. 2006) is a free standalone tool for detecting similar passages in large collections of students' coursework. It enables large numbers of documents to be analyzed quickly, and can also be used to identify plagiarism. The Ferret copy detector works on phrase-based mechanism to determine the similarity

between two documents. Usually, the results are presented in a ranked table with the identical or most similar pairs at the top. Bao et al. (2006) used Ferret for copy detection in Chinese documents. Corpora of students' coursework from two Chinese universities were collected, and Ferret was applied to investigate the detection of plagiarism. Experiments showed that Ferret can find plagiarism inChinese documents efficiently.

The survey on plagiarism detection for Arabic language that has been reviewed. We organize a table to explain a thorough survey of state-of-the-art plagiarism detection techniques and to better understanding we produce some charts based on our literature review statistics. Most techniques detect plagiarism by using certain text features along with fingerprint matching techniques and most of the them used some algorithms in the pre-processing stage of the system like normalization, tokenization, stemming and part of speech (POS) tagging, stop-word removal, sentence segmentation, synonymy recognition, number replacement, lemmatization. It is obvious that all utilized techniques are showed in the table 2.2 has its own impact on developing plagiarism detection for Arabic Language. Most of the studies and developments are stretched in literal type of plagiarism while the minor works dealt with intelligent type. A few numbers of study produced an implemented tool or software meanwhile the others proposed a development in a particular algorithm or technique, the summery of each study that have reviewed are explained in table 2.2.

Table 2.2: Extracted Papers Based on the Criteria

Ref.	Type of Criteria	Source or target	Year	Language	Techniques	Result
[48]	intelligent	Document	2009	Arabic	Fuzzy technique in information retrieval	Stated that Fuzzy technique is better than Boolean IR ,in plagiarism detection
[50]	literal	E-learning	2009	Arabic	Syntax Similarity based detection	For the first time created APD tool for Arabic in e-Learning.

[18]	Literal	Text	2010	Arabic	fingerprint matching	Improved fingerprint matching technique through Adding four key features of the text.
[19]	Literal	Document	2011	Arabic	Fingerprinting	APlag, a plagiarism detection tool for Arabic language.
[51]	Literal	Text	2012	Arabic	Stylysis tool.	Discover the effect of some well-known language-independent stylistic features on Arabic text to improve Plagiarism detection.
[74]	Literal	Text & Document	2012	Arabic	winnowing n-gram fingerprinting	It proposed mono-lingual system (Iqtabs 1.0) for plagiarism detection that precedes multi-lingual
[17]	Literal	Document	2012	Arabic	Fingerprinting and Similarity metric	Improved Aplag
[52]	Intelligent	Text	2013	Arabic	Examined the existing literal systems.	It presented a new taxonomy of plagiarism that highlights differences between literal and intelligent plagiarism. They emphasized that existing systems for intelligent plagiarism detection are failed.

[53]	Literal	Authorship	2013	Arabic	Word N-Grams.	Stated that good attribution performances with an optimal score of 80% of good authorship attribution
[54]	Literal	Authorship	2014	Arabic	MBNB technique Naïve Bayes classifiers	Attribute the author of a text with an accuracy of 97.43%.
[55]	literal	Authorship	2014	Arabic	Two popular classifiers: FT and SVM.	Stated that the FT method has better performance as Accuracy of 82% was achieved.
[10]	literal	Document	2015	Arabic	Similarity technique in information retrieval	A web-based plagiarism detection framework for Arabic documents.

All these practices of plagiarism have a negative impact on the learning process. Thus, how can we ensure dealing with Plagiarism systems and how is plagiarism going to detected. A critical issue needs solutions by computer scientists. [25]

2.10 Arabic Plagiarism Detection Systems

The interested reader may refer to a number of surveys on the subject of detecting plagiarism in the year and in other languages, but we will focus on the Arabic languages [83], [80], [81] and [79]. In the statement of Arabic language, several plagiarism detection systems are proposed. For instance, Alzahrani and Salim [23] have introduced a statement-based plagiarism detection system for Arabic (FS-APD) using

fuzzy-set information retrieval model [82]. The degree of similarity between two statements is computed and compared to a fixed threshold value to judge whether are similar or not. This approach led to perform well on verbatim reproductions. To address the rewording, they have proposed another system named fuzzy semantic-based string similarity for extrinsic plagiarism detection (SFS-APD) [84]. This uses a shingling algorithm, Arabic WordNet lexical database [77] and Jaccard coefficient for retrieving a list of candidate documents. The suspicious document is then compared sentence by sentence with the candidate documents to compute the fuzzy degree of similarity.

Jadalla and Elnagar [2] introduced a plagiarism detection system for Arabic text-based documents named Iqtebas. It uses a fingerprint search engine to compute the distance between each sentence in the suspected text and the closest sentence in the source documents. Iqtebas seems to perform well the copy-and-paste (C&P) plagiarism, but it handles neither word shuffling nor rewording.

Recently, Hussein [85] proposed a new plagiarism detection system for Arabic documents based on modeling the relation between texts and their n-gram unique sentences. The system involves several steps, including Part-of-Speech (POS) tagging, text indexing, stop-words removal, synonyms substitution and heuristic pairwise phrase matching algorithm to build documents Term Frequency-Inverse Document Frequency (TF-IDF) model [89]. The Latent Semantic Analysis (LSA) [90] and Singular Value Decomposition (SVD) are then used to analyse the hidden associations between text documents. [91]

The Arabic Plagiarism Detection Shared Task 2015 (AraPlagDet)2 [16] is the first and only shared task that addresses the evaluation of plagiarism detection methods for Arabic texts. It has two sub-tasks: extrinsic and intrinsic plagiarism detection. A major advantage of the AraPlagDet evaluation campaign is enabling the evaluation of different systems on the same dataset. In AraPlagDet 2015 three systems are participated in the extrinsic plagiarism detection subtask: Magooda [86], Alzahrani[87]

and Palkovskii3. Two participants (Magooda and Alzahrani) among the three submitted working notes describing their systems.

Magooda et al. [86] proposed an extrinsic plagiarism detection system named RDI_RED. In this system, Lucene search engine [88] is used to select a list of candidate source documents. The candidate documents are aligned to detect plagiarised segments (aligned parts). Finally, a set of rules is applied by a filtering module in order to filter the aligned parts. RDI_RED system can be easily deployed on-line. Though, it does not address synonyms substitution and paraphrasing. [88]

Alzahrani's [84] introduced system goes through four main steps. The first step pre-processing, this includes tokenization and stop-word removal. In second step, retrieve a list of candidate source documents for each suspicious document using n-gram fingerprinting and Jaccard coefficient, the third step an in-depth comparison between the suspicious documents and the associated source candidate documents using k-overlapping approach [79], in final step Post-processing where consecutive n-grams are joined to form united plagiarised segments. Table 2.3 summarizes the Arabic plagiarism detection systems described above according to the technique used, the comparison level and their efficiency in detecting different plagiarism types. [79]

Table 2.3: Details of the Arabic plagiarism detection systems

		FS-APD [90]	SFS-APD [89]	Aplag [11]	Iqtebas [2]	Hussein [85]	RDI-RED [86]	Alzahrani [84]
Technique	Fingerprinting			*	*			*
	Fuzzy-set	*	*					
	SVD					*		
	LSA					*		
	Search Engine						*	
	Linguistic Resources		*	*		*		

	Word Embedding							
Comparison Level	Sentence-Level	*	*	*	*	*	*	*
	Paragraph –Level			*			*	
Plagiarism Type	C&P	*	*	*	*	*	*	*
	Reordering	*	*	*	*	*	*	*
	Synonyms Substitution		*	*		*		*
	Paraphrasing							*

Our plagiarism detection tool built around a content-based method. It fulfills the three properties. The first property is to handle by a preprocessing of any input text, including tokenization, stop-word removal, rooting and synonym replacement. It is constracted on fingerprinting 3-grams of chunk. The second property is satisfied if 3 is sufficiently long to ignore common idioms of Arabic language. The third property is can demonstrate by the performance results on the datasets.

2.11 Summary

To sum up, the literature review has been investigating in Plagiarism definition and Types, Way and strategy to Avoid Plagiarism .characteristics of Arabic language, Plagiarism in Arabic documents fingerprint matching technique, Plagiarism Techniques and Algorithms, Plagiarism Detection tools for natural language Documents summarization of Arabic Plagiarism Detection Systems.

CHAPTER III

ARABIC DOCUMENTS PLAGIARISM DETECTION MODEL

3.1 Introduction

This chapter deals with the concepts and terminology of the main model components for Arabic documents plagiarism detection. It starts with overview of main model and goes deep on details.

3.2 Arabic Documents Plagiarism Detection Model

Figure 3.1 Shows the Main components of the introduced Arabic documents plagiarism detection model. These components are shown below.

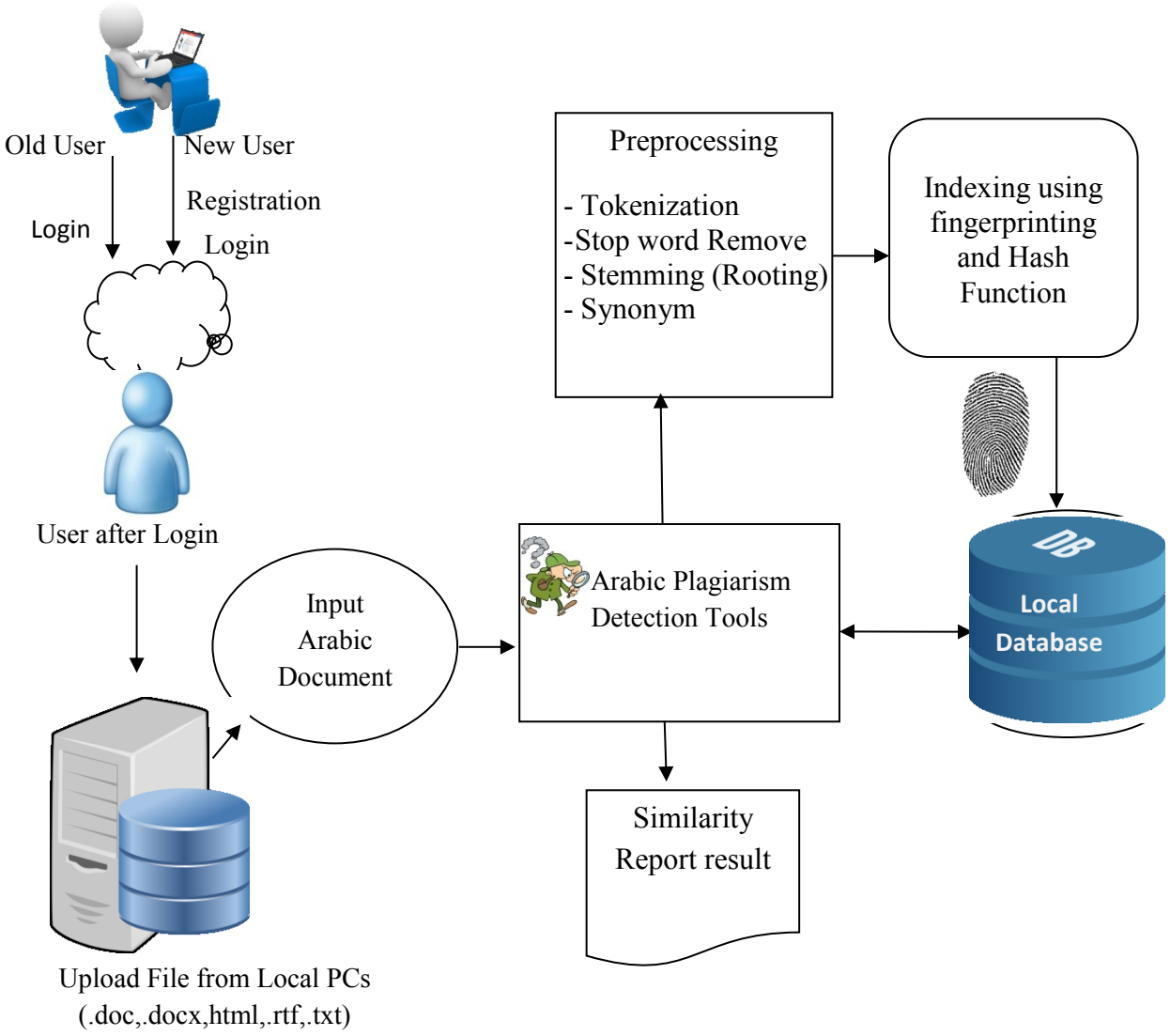


Figure 3.1: The Main components of Arabic documents plagiarism detection Model.

Figure 3.1 depicts the overall processes and components of the proposed Model. The model consists of five stages. The first stage files upload and conversion. If the file in that formation (.doc, .docx, html, .rtf,.dot) then it will converted to txt format (.txt) this important issues. Second stage is the text pre-processing, which consists of documents, tokenization, stop-words removing and word stemming. The aim of the three stages is to convert the output of the previous stage to fingerprint using n-gram method. The four stage is to save the fingerprint for each document .the five stage is to fine the similarity matching between the input text with local database and gives similarity detection report.

3.2.1 Details Pertaining to Arabic Documents Plagiarism Detection Model

The details pertaining to Arabic documents plagiarism detection mode as shown in Figure 3.2.These details pertaining is components describe as follows:

3.2.1.1 Preprocessing

In this section, the preprocessing is a core natural language processing task. It aims at creating an intermediate form from the inputted text based on the extraction of words, the morphological analysis, and the text annotation. The researcher adopts detection "Content-based" as primary treatment in which the removal of stop-words developed and lowered words to form roots. Following stages are perform to transform the Arabic text to organize and formatted of the representation, which is more suitable for the process of detecting plagiarism. Following stages are perform to convert a document in Arabic, to build and prepare represented that it is more agreement for the processing of detecting plagiarism. It is handling by a preprocessing of any input document, including tokenization, stop-word removal, rooting and synonym replacement.

A. Tokenization

The stream of Arabic text divided into words, phrases, symbols, or other meaning parts. The list of tokens inserted input to next pre-processing steps.

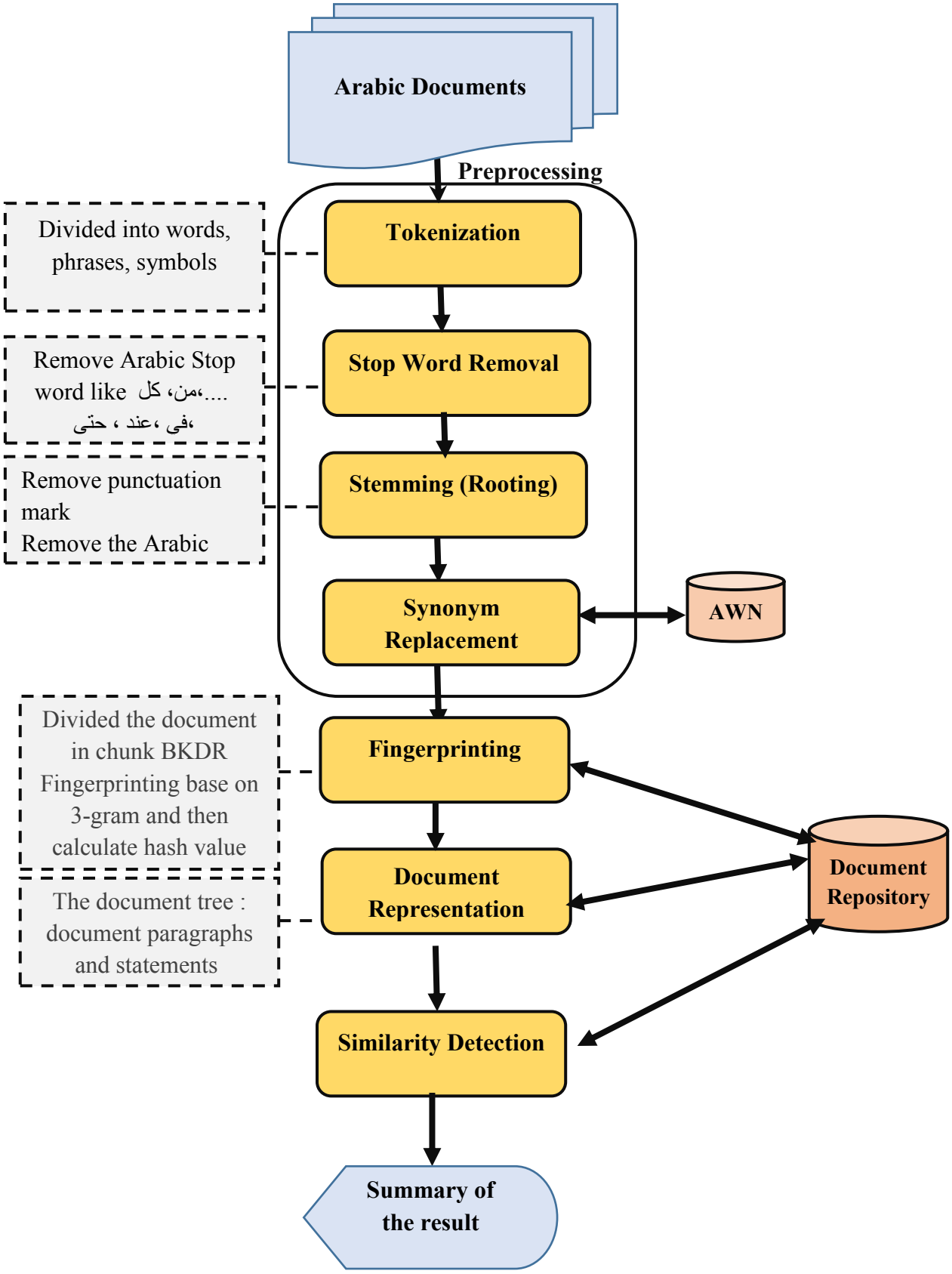


Figure 3.2: Details pertaining to Arabic Documents Plagiarism Detection Model

B. Removing Stop Words

Arabic language has high inflections and eloquence that build words not significant for the retrieval process because they are redundant and do not influence the meaning. These words named stop words and may exist in both query documents and corpus collection.Stop words are exclude words, these words are to exclude by language automated processing of data (texts). These words are repeating in the texts, which are contains 162 words like (من، حتى ، عند ،في ،كل...) normal and 1062 extented [4]. It is advisable to be removed form document and not indexed in order to improve the search. Thanks to Hans Peter Lohan (of the pioneers in information retrieval) in the use of the term and concept development. These words do not give these words do not give any hint values or meanings to the content of their documents, hence deleted words from the set of index terms [4]. Omission the stop words in automatic indexing is speed of system process, saves a huge amount of space in the index, and does not damage the retrieval effectiveness [39]. For example, "يذهب أحمد إلى المدرسة كل يوم بصحبة صديقه عمر" becomes "يذهب أحمد المدرسة بصحبة صديقه عمر".

C. Stemming Words (Rooting)

Stemming is a process of remove the affixes. The affixes is contains prefixes, suffixes and infixes. The prefixes "are a group of words attached at beginning of the word". In addition, the suffixes "are attached to the end" .and the infix "is found in the middle of word". (morphemes) in a word in order to generate its root word as Khoja's stemmer [76].as showing in table 3.1 and figure 3.3 is an example how to extract root. Using the root word in pattern matching provides a much better effectiveness in information retrieval. There are several stemmers existing in the Arabic, English language, such as Nice Stemmer, Text Stemmer and Porter Stemmer are the well-known English stemmer that commonly been used[76]. Figure 3.4 shows the Arab sentences and steps preprocessing in the introduce plagiarism detection model.

After remove stop-word, punctuation and delete the numbers, spaces and single letters, then Convert letters (ء), (ؤ), (ئ), (أ), (إ) into (ا), and (ة) into (ه). Novelty

the basic root of Arabic words by removing affixes (suffixes and prefixes) attached to its root. Prefix like (ال, وال, بال, كال، فال, لل) and suffix like (ها,ان, ية,يه, ات, ين,ون, ه, قو, ي) table 4.1,and table 4.2.

Table 3.1: an example of the Arabic Affixes stemming

Word	Root	Prefix	Suffix	Infix
الخالدين	خلد	ال	ين	ا

Figure 3.3 Arabic words extract rooting

C.1 Rules to Remove the Affixes

Removing the determiner "ال": when remove the determiner "ال" and its combinations. All these characters must remove from the word, since these letters are the leftmost prefixes that can appear in an Arabic word. Before removing any prefix or suffix, the algorithm checks the size of the word; the number of characters remaining word length must be greater than or equal to 3. For example, the prefix " بال " does not remove from the word" بالغ ". Some words have these same characters as root characters (e.g. " بالغون ", "فالحين" and "كالحون"). To stem such words correctly we check these patterns before removing their prefixes. Using this rule the word" بالغون ", for example, will reduce to the word" بالغ ", as we will explain later and then return the stem" بلغ ".

Removing prefixes: The next step is to remove all multi-letter prefixes that have no duplicated. If these letters found then the first one are considered a prefix and will remove. For example, the words"ككتاب ", " نتبع " and " ووادي "will be reduced

to "كتاب", "تبع" and "ولدي", respectively, Arabic stemmer rules do not check the single letter prefixes ("ي" and"ت") because these characters could be root letters and not prefixes. For example, the letters "ت" and "ي" in the words "توبة" and "يومه", respectively both belong to the stem. So after removing the suffixes later, the remaining word will be retained as a stem since its length is 3.

Removing suffixes: word must reduce in order to match an appropriate pattern. Therefore, the inflected word enters this step, the algorithm checks for the suffixes working from the longest to the shortest one. As mentioned above, the algorithm checks the length of the word before removing any suffix; the length of the remaining word must be greater than 2.

Removing "ف" and "و": These two letters have the meaning of (then) and (and) in English respectively, so they written before any single letter prefix as "ي", which indicates the present form of the verb, but in Arabic they cannot be used together and still have the same meaning. Therefore, if both of them appear, the second letter will not be a prefix. In this step, stemmer checks one of them only. These letters can sometimes be root letters not prefixes, for example: "واحد", "فارس", "فعّال" etc. it is difficult to distinguish these words without using a database containing all Arabic stems. To resolve this ambiguity we use some rules that depend on patterns. If the word matches a certain pattern, then the letter not removed.

Although this technique resolves this problem partially, it sometimes fails with some words, especially when two words reduce to the same string. For example, consider the pair of words "وقول" and"ورود", the letter "و" is a prefix in the first word but not in the second one.

D. Arabic Synonym Replacement

The words were regenerate to their most frequent synonyms, which can facilitate to notice advanced varieties of hidden plagiarism. Word synonyms area unit retrieved from

Arabic WordNet (AWN). The primary word within the list of synonyms of a given word is taken into account because the most frequent one.

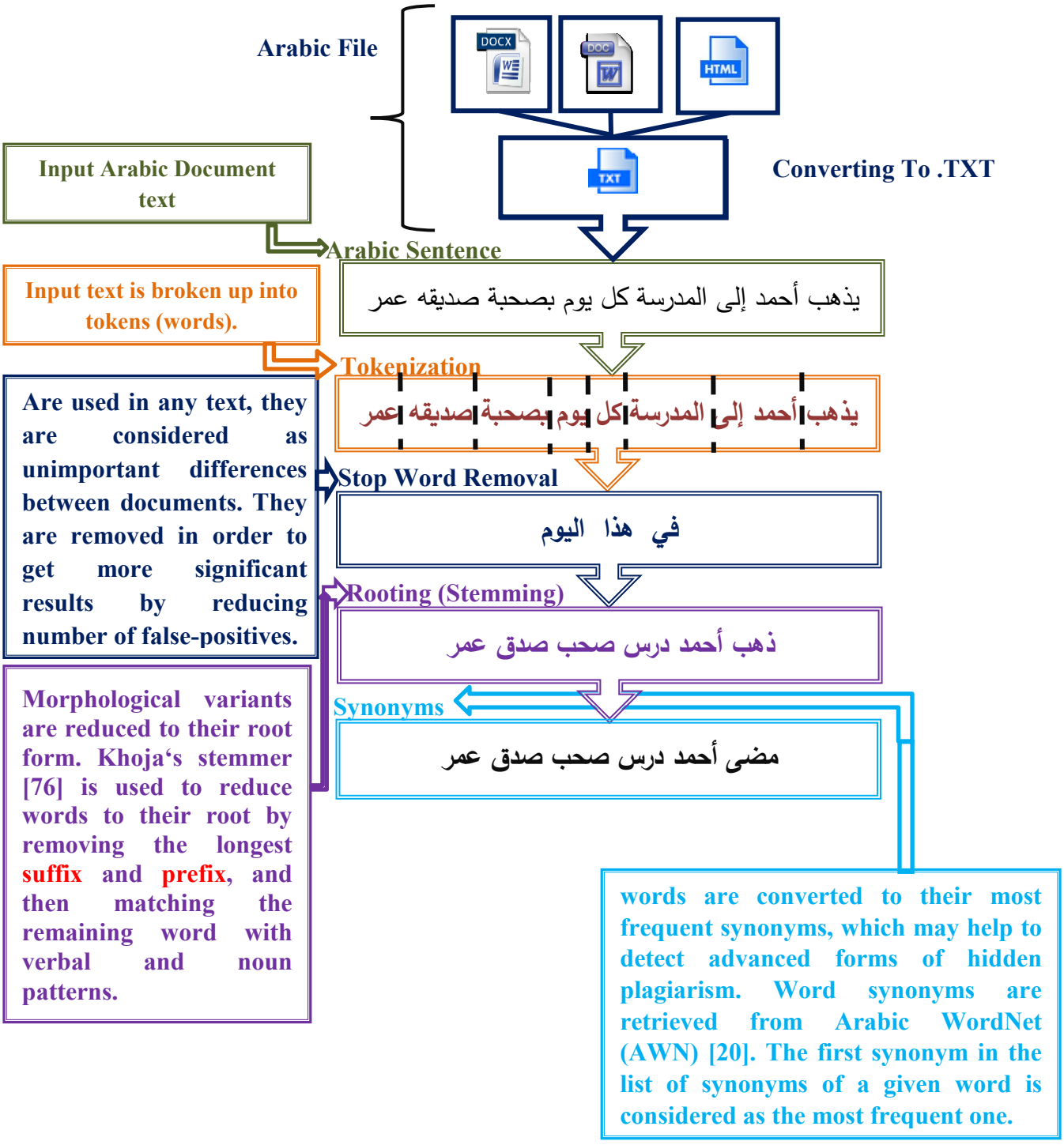


Figure 3.4: An example of Arabic sentence preprocessing steps

3.2.1.2 Fingerprinting

The fingerprint matching technique is widely used in the plagiarism detection tools. The main idea in the document fingerprinting to detect the reuse of the text is to generate a numerical representation of the unique document (in the case of disclosure of the exact copy), or part of the text (in the case of partial / detectors local copy). Then, (body) will be used for these assertions in a document candidate comparison against a set of documents. The process of creating a fingertprint consists of four main steps [30]: the first is the function that generates hash-value from a substring in the document. The second is the granularity; that is the size of the substring that is extracted from the document (chunk size). The third is the resolution which is the number of hash-values used. The fourth is the strategy that is used to select substrings from the document[29]. Included two of the main characteristics that a good technique fingerotprint should meet: generate fingerprints that accurately represent documents, and produce the least number of possible fingerprints fingers.[29]

3.2.1.3 Document Representation

The tree structure of the Arabic document is created to describe the internal representation of the documents. Every document created to represent a tree to describe the document's logical structure. The same document contains the root, and the second level contains the vertebrae and the leaf nodes contain sentences. Figure 3.5 representation document tree appears. This representation is links to those used in the verification of [13], and plagiarism detection system. It is consider avoiding comparisons unnecessary among several documents. The establishment of a tree representation of each document is then explored the trees from top to bottom, and compared to the level of the level until a termination condition. [14]

3.2.1.4 Comparison of Similar Term

In this stage, heuristic Algorithm is used to find the longest match of two hash strings by similarity method. In comparison at the document level scope, we compared two documents in accordance with common hashes and their fixed threshold. If the number of partitions in a subset of a larger crosses the threshold, then there is a possible

similarity between the two documents. In this case, the comparison process is still at the paragraph level, is detected any similarity is shut down the operation. If the detection probability of similarity to the paragraph level, and then the process will continue on the wholesale level, otherwise the process terminates. In case of similarity between two sentences, then use longest common substring (LCS) to measured using the metric. Uncertainty the length of the longest corporate sequence is greater than the length multiplied by the minimum sentence threshold, then they determine similar chains in each of the strings, but this process will continue with the following sentence. We use a heuristic algorithm of each level of the tree base on the document, paragraph and sentence.

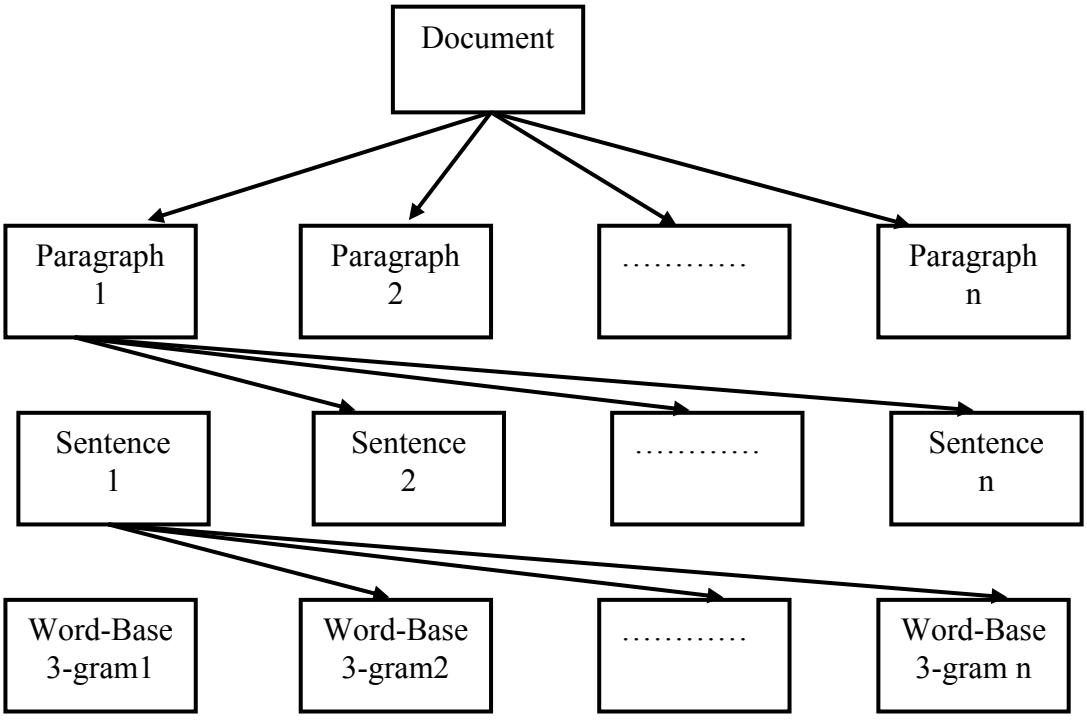


Figure 3.5: Arabic document tree representation

In the event chunking, the similar chunks found in document of sentence-based, and then we divided based of parts parameter n, which will be grouping into the form of sequence of n sentences into a chunk. In case of Word-based chunking gives higher accuracy in detecting similarity than sentence-based chunking [21] .It is important to choice a hash function that reduces collisions due to mapping different chunks to the same hash. Our methods based on a word-based chunking method: in every sentence of a document, words are first chunked and then hashed using a hash function[22].For

example, given a document containing the sentences *se1 se2 se3 se4 se5*, if n=3 then the chunks are *se1 se2 se3, se2 se3 se4, se3 se4 se5* [21]. Another example of word, given a document containing the words *wo1 wo2 wo3 wo4 wo5*, if n=3 then the chunks are wo1 *wo2 wo3, wo2 wo3 wo4, wo3 wo4 wo5*.There are some strings matching, algorithm transforms one string another. levenshtein distance (LD) and Longest Common Substring (LCS), those algorithms are measures the minimum number of operations: insertions, deletions, or substitutions to transform one string to another[23]. Consists in finding the common longest substring in two strings. Let us consider a longest substring to check in "الخالدين" and "الوالدين" is "دين" For plagiarism detection, if the plagiarism or similarity ,the LD and LCS are more appropriate , because similarity requires modification of a text . In our approaches we considered the LCS, because we believe to use LCS, because it is based on the phenomena of similarity rather than distance.[24]

3.3 Summary

To summarize, depicts the overall processes and components of the proposed model and details pertaining for Arabic documents plagiarism detection, which is consists of five stages, files upload and conversion, The second stage is the document pre-processing, the three stages is to generate BKDR fingerprint using 3-gram method for the document. The four stage is to save the fingerprint for each document .the five stage is to fine the similarity matching between the input text with local database and gives similarity detection report.

CHAPTER IV

PLAGIARISM DETECTION FRAMEWORK AND TOOL

4.1 Introduction

This chapter presents the operational framework for plagiarism detection framework and tool.

4.2 Operational Framework

This chapter was conducte according to the workflow process illustrated in Figure 4.1. The operational framework is divided in to six phases: starting from The starting from planning phase until summary report. The planning and preprocessing stage include planning the research and reviewing the previous work, building the corpus collection, proposed plagiarism detection framework includes of four main Phases. In this first phase, upload Arabic file, second phase Preprocessing, third phase Indexing and Hashing, and the fourth phase Similarity Matching. We focus on detecting the Arabic - Arabic plagiarism. As a plagiarism detection system, our corpus builded up the Internet resources that are detectable by the AraPlagDet share task 2015. Figure 4.1 shows framework Arabic language plagiarism detection.

4.2.1 Planning Phase

In the planning phase, literature search of Arabic document plagiarism hdetection as been done in order to benefit from the previous efforts of the pre-processing steps such as removing stop words and stemming Arabic words. In addition, literary research on plagiarism detection techniques applied to English, which was not use in Arabic, have explored in order to select the most appropriate, efficient and useful methods for use in the detection of plagiarism in Arabic.

4.2.2 Building Corpus Collection

The corpus for this study will use initial data building our self and InAraPlagDet-20-06-2015 on AraPlaDet browser and Wikipidia with 1036 documents

chosen arbitrary about different topics including Create your own country blog, Islamic book, Corpus of Classical Arabic and DSS.

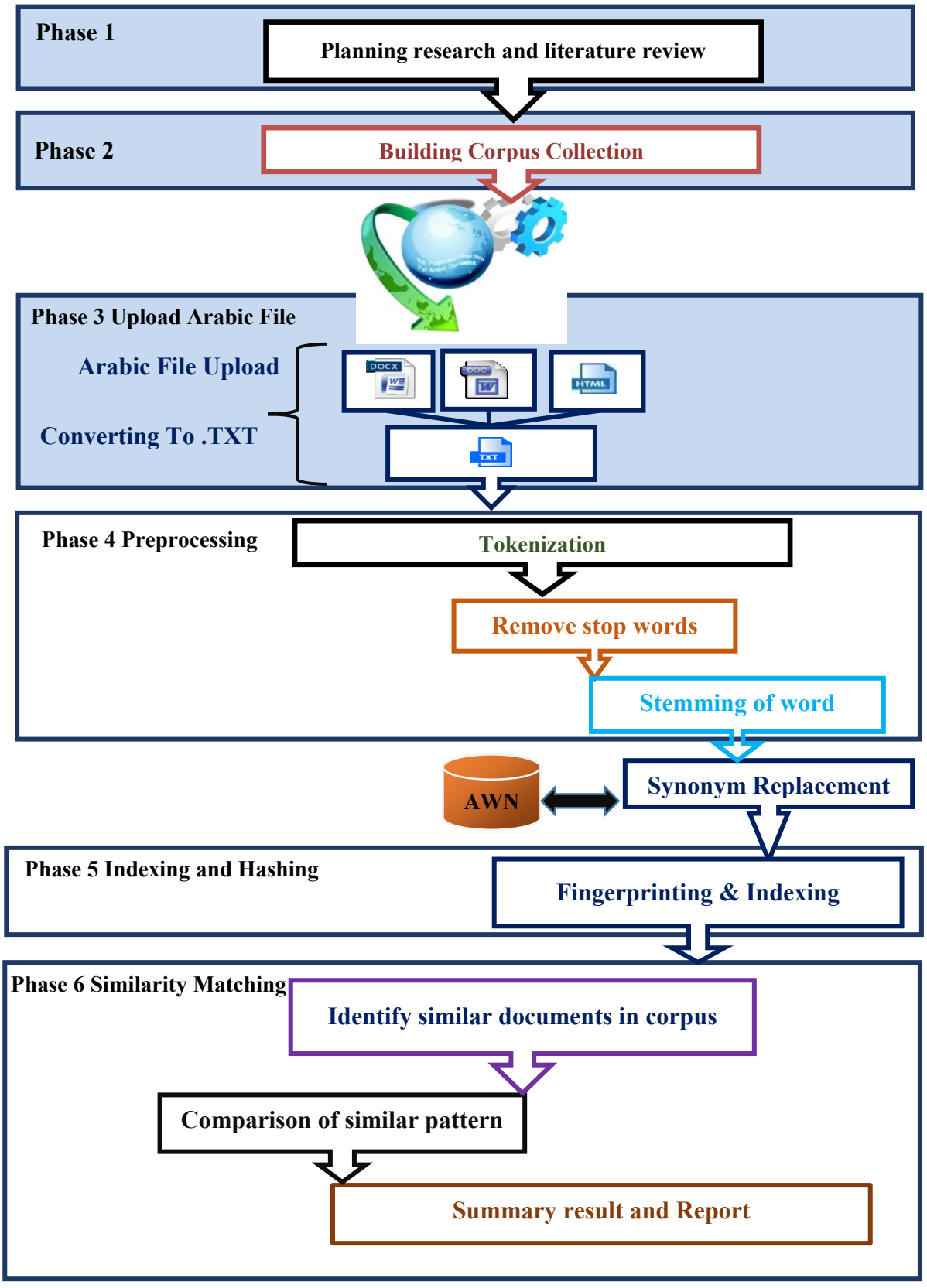


Figure 4.1: Flow chart of the Framework for Arabic Document

Moreover, we preferred to save the corpus documents in TXT file format using UTF-8 encoding. Various browers support UTF-8 and it is unnecessary to set-up a language encoding. These advantages make UTF-8 encoding practical, and of much help in our case, to support bilingual documents (Arabic) since we will use PHP in developing and testing the techniques.

4.2.3 Input Documents

In this step we accept file include (.doc, docx, html, .txt). Before we use as the query documents for further detection process. The files [(.doc, docx, .html] must convert to .txt format. For example, "يذهب أحمد إلى المدرسة كل يوم بصحبة صديقه عمر" this sentence can in different formation file extension.

4.2.4 Tokenization

The stream of Arabic text will divided into words, phrases, symbols, or other meaning parts. The list of tokens becomes to input for next preprocessing step.

4.2.5 Removing Stop Words

Stop words are excluded words are words that are excluded by language automated processing of data (texts). It is words that repeated in the texts, which are contains 162 words like (من، حتى ، عند، في، كل...) It is advisable to be removed form document and not indexed in order to improve the search.

4.2.6 Stemming (Rooting) Proceeding

Arabic words demonstrate an intricate morphology[4]. The Arabic language can be said to use root-and-pattern morphotactics where a pattern can be thought of as a template adhering to established grammatical rules.

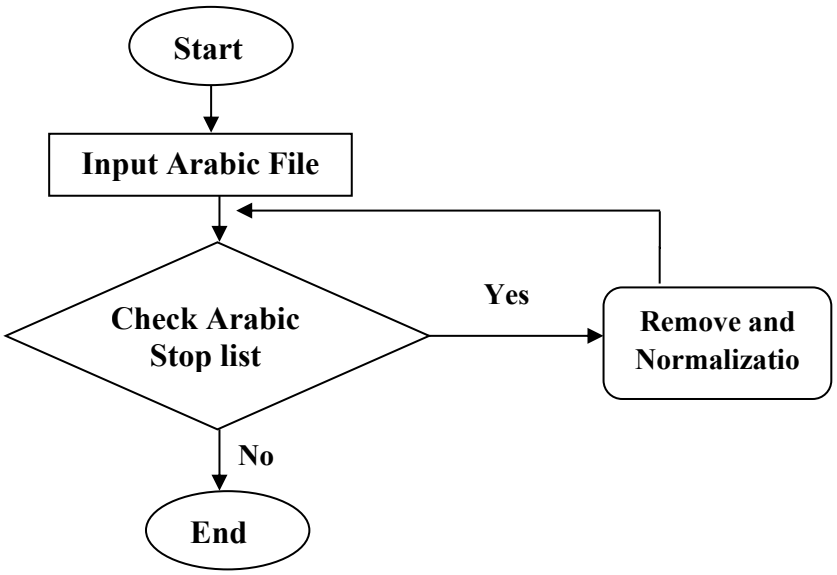


Figure 4.2: Arabic stop word list removable process

In this example, will explain how to extract the word roots as showing in figure 4.5, (which are simple bare verbs that are three letters in length) to form their parent root by an verb weight from the word ("الخالدين") that mention in table 3.1 chapter3. For rooting (stemmer process) is (خلد), so will go throw the process below. The mechanism begins by receiving word by word from document A ~ and then entering the first test. Is the word found in the Arabic dictionary list, if it matches a reservation in another file, (if not matching prefix, suffix and infix if found A ~~). so "الخالدين" Is not among the words in the dictionary and then enter the test (Prefix List) to remove from the list of prefix they rules that mentioned above so the determiner "ال" is removed returning "خالدين" ,then no prefix are found and then enter the test of (Suffix List) will determiner "ين" are founded in the list will removed returning " خالد" then no suffix are found and then enter the test of (infix List) will determiner "ا" are founded in the list will removed returning "خلد".

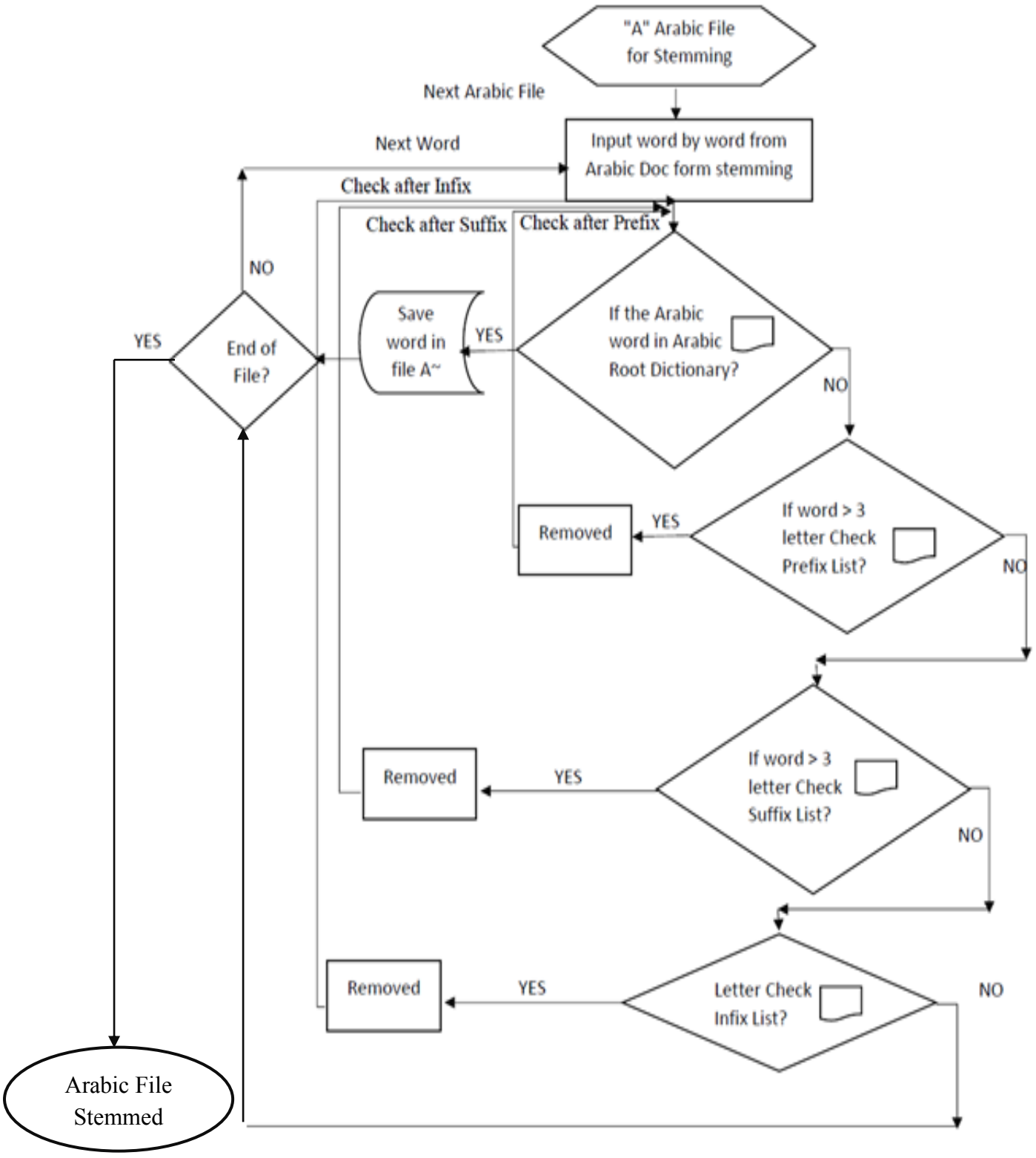


Figure 4.3 : Arabic stemming (root) process

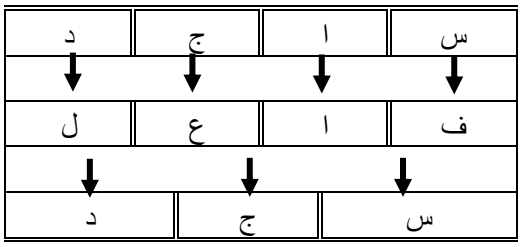


Figure 4.4:Extracting the stem of the word "ساجد" from the pattern "فاعل" Arabic stemming

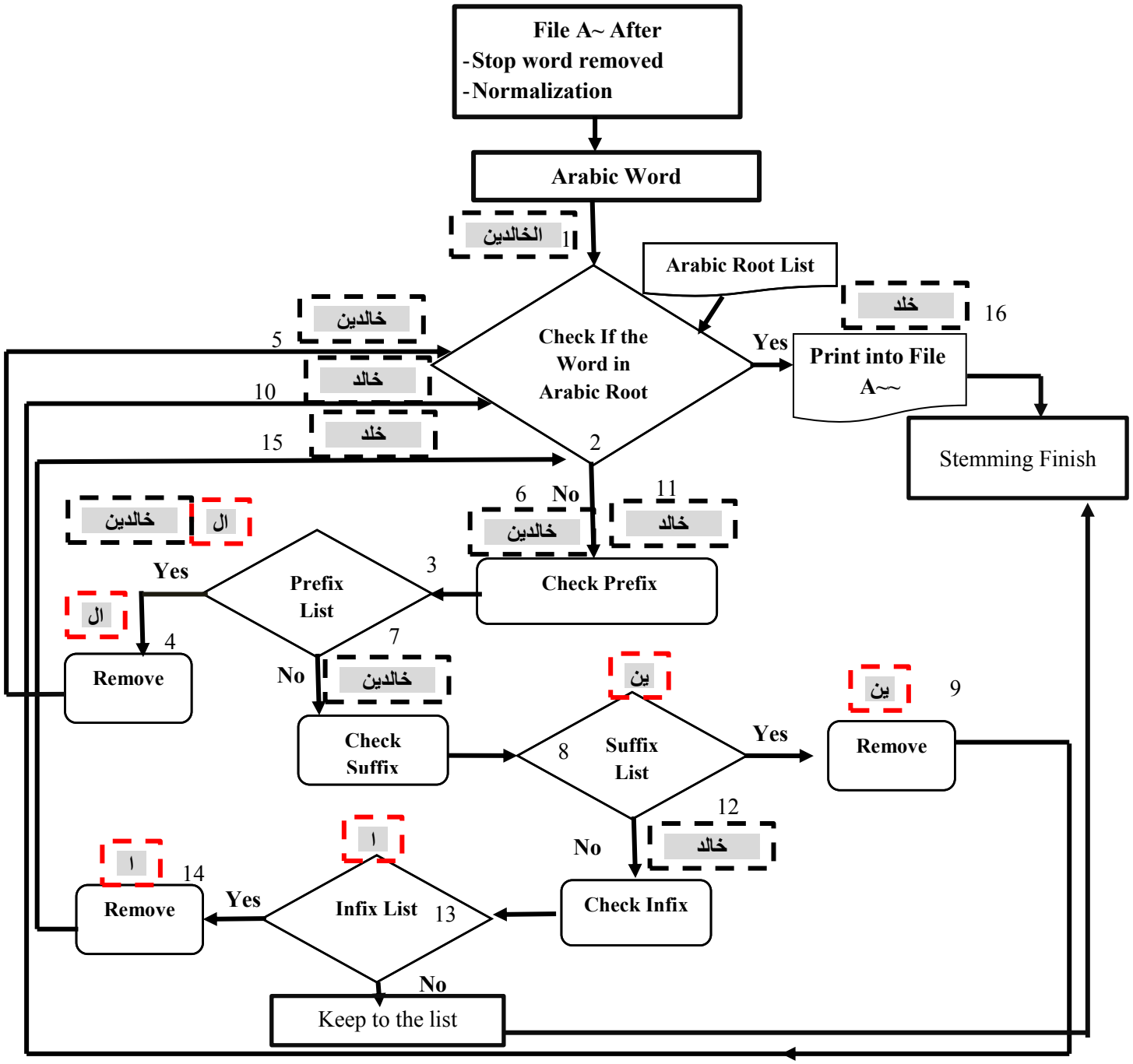


Figure 4.5: Example of an Arabic(الخالدين) Stemming (Root) word process

Table 4.1 : Arabic prefixes

Prefix	Example
ب	بالسيارة
ك	كالدخان
س	سأذهب
و	ورجالهم
ال	النساء
أ	أأكلت
ف	فذهبوا
ل	لتنام
ت	تلعب
م	مستخلفون
ن	نتوكل
ي	يتوكل
لل	للكرة

Table 4.2: Arabic Suffixes

Suffix	Example	Suffix	Example
ية	التراثية	ين	تلعبين
نا	صحبنا	ان	تلعبان
تموها	نسبتموها	و	ينمو
هن	هديلهن	ه	ضربته
كما	بكاؤكما	ة	خبرة
كن	حسبكن	ك	ضربك
هنن	أخواتهنن	ا	أكلا
ني	أراني	ي	أكلتي
اتي	حساباتي	ن	أكلن
		ت	أكلت
		ات	لاعبات
		ون	يلعبون
		وا	أكلوا
		تم	أكلتم
		هم	ضربهم
		كم	ضربكم
		اء	ميساء
		هما	أكرمهما
		يا	النوايا
		ها	منزلها
		وها	سنلزموكموها
		يه	والديه
		ء	الرمضاء

4.3 Fingerprinting Process

The fingerprint matching technique is widely used in the plagiarism detection tools. The main idea in the document fingerprinting to detect the re-use of the text is to generate a numerical representation of the unique document (in the case of disclosure of the exact copy), or part of the text (in the case of partial / detectors local copy). Then, (body) will used for these assertions in a document candidate comparison against a set of documents [1, 41]. The process of creating a fingerprint consists of four main steps [2]: the first is the function that generates a hash-value from a substring in the document. The second is the granularity; that is the size of the substring that was extracted from the document (chunk size). The third is the resolution, which is the number of hash-values used. The fourth is the strategy that used to select substrings from the document. [41]

A. Document Representation

As shown in figure 3.5 in chapter 3, the stem consists of the tree basic document, the second level consists of all refined text paragraphs, and the third level of the tree encompasses the sentences of the paragraph. The tree structure of the Arabic document is created to describe the internal representation of the documents. Every document created to represent a tree to describe the document's logical structure. The same document contains the root, and the second level contains the vertebrae and the leaf nodes contain sentences. Figure.3.5 representation document tree appears. This representation is links to those used in the verification of [13], and plagiarism detection system. It is consider avoiding comparisons unnecessary among several documents. The establishment of a tree representation of each document is then explored the trees from top to bottom, and compared to the level of the level until a termination condition. [14]

Then sentences are divided into word-based 3-grams, and using a proper hash function, they are converted into a number. In this manner, the processing speed is increased in the copy detection operation. In figure 4.6, there is a tree representation of the single sentence paragraph "طقس نجران اليوم غائم وممطر.".

Document text :	طقس نجران اليوم غائم وممطر.
Preprocessing :	طقس نجران غيم مطر
Paragraph Level:	طقس نجران غيم مطر
Sentences Level:	طقس نجران غيم مطر
Word-based 3-gram level	نجران غيم مطر طقس نجران غيم

Figure 4.6 Arab document preprocessing base on 3-gram

It is important to select a hash function that minimizes the collisions due to mapping different chunks to the same hash [6, 10]. In this implementation, the BKDR hash function is used. This function is the sum of each character's multiplication in a certain value named "seed" that usually has the value of 31. The seed value must be an odd number because odd numbers are unique, and multiplication of a number in an odd number creates a unique hash value as shows in eqation (1) [6, 10]. The steps for the above example of fingerprinting are shown in figure 4.7. The fingerprint of this single sentence paragraph is 937118507.

$$\textbf{Hash value} = \boldsymbol{s}[\mathbf{0}] * \mathbf{31}^{\boldsymbol{n-1}} + \boldsymbol{s}[\mathbf{1}] * \mathbf{31}^{\boldsymbol{n-2}} + \ldots + \boldsymbol{s}[\boldsymbol{n-1}] \quad (1)$$

Using int arithemrtic, where s[i] ith ith character unicode of the string, n is the lenght of chunck ,$^{n-1}$ is indecates exponentiation.

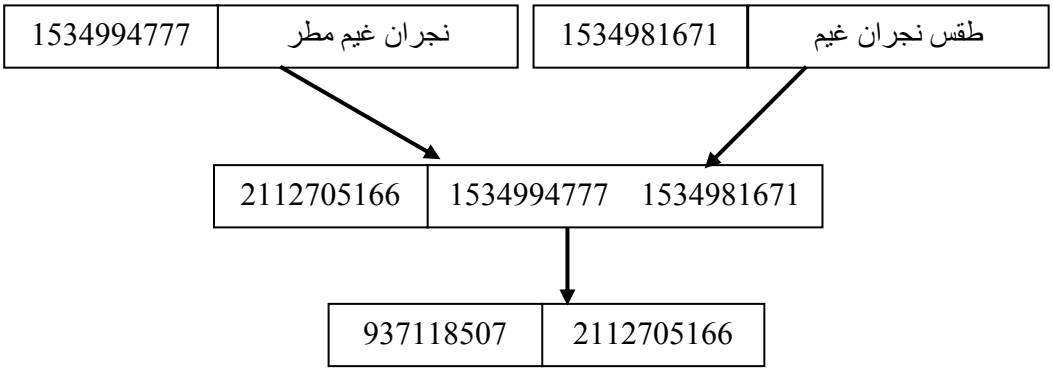


Figure 4.7: Arabic Document Fingerprinting example

According to figure 4.7, after breaking all the words contained in sentences into 3-grams, it is time to hash operations at sentence-level. Through this procedure, the hashes obtained from words-based 3-grams are broken into 3-grams in tree sentence-level, and a hash operation is run on them. In the final step, the hashed 3-grams will be converted from sentence-level into paragraph-level 3-grams. Therefore, the document fingerprints obtained contain paragraph-level hashes of the document.

4.4 Comparison of Similar Term

Many similarity metrics exist for fingerprint comparison, including Levenshtein distance [23], Longest Common Substring (LCS), and Running Karp-Rabin Matching and Greedy String Tiling (RKR-GST) [23]. The Levenshtein distance measures the minimum number of operations: insertions, deletions, or substitutions to transform one string to another. For example, the Levenshtein distance between "Saturday" and "Sunday" is three. The Longest Common Substring (LCS) consists in finding the common longest substring in two strings. For example, the common longest substring in "Saturday" and "Sunday" is "day". RKR-GST [24] is use for comparing amino acid bio-sequences. It consists in tiling one string with matching substrings of a second string. RKR is an improvement technique to speed up the GST algorithm. A hash value is created for each substring of length s of the pattern string and for each substring of length s of the text string. Each of these hash values of the pattern string is compared with the hash values of the text string. If the pattern and text hash values are equal, then there are matches between the corresponding pattern and text substrings. A key issue in similarity detection is to choose the adequate metric. For plagiarism detection, Levenstein distance and LCS are more suitable, since plagiarism involves modification of a text (insertion, removal …). In ADPDM, we choose to use LCS, because it is base on the concept of similarity rather than distance [41].

In the event chunking, the similar chunks found in document of sentence-based, and then we divided based of parts parameter n, which will be grouping into the form of

sequence of n sentences into a chunk. In case of Word-based chunking gives higher accuracy in detecting similarity than sentence-based chunking [21], .It is important to choice a hash function that reduces collisions due to mapping different chunks to the same hash. Our methods based on a word-based chunking method: in every sentence of a document, words are first chunked and then hashed using a hash function[22].For example, given a document containing the sentences *se1 se2 se3 se4 se5*, if n=3 then the chunks are *se1 se2 se3, se2 se3 se4, se3 se4 se5* [21]. Another example of word, given a document containing the words *wo1 wo2 wo3 wo4 wo5*, if n=3 then the chunks are *wo1 wo2 wo3, wo2 wo3 wo4, wo3 wo4 wo5*.There are some strings matching, algorithm transforms one string another. Levenshtein distance (LD) and Longest Common Substring (LCS), those algorithms are measures the minimum number of operations: insertions, deletions, or substitutions to transform one string to another [23]. Consists in finding the common longest substring in two strings, let us consider a longest substring to check in "الخالدين" and "الوالدين" is "دين" For plagiarism detection, if the plagiarism or similarity ,the LD and LCS are more appropriate , because similarity requires modification of a text . In our approaches we considered the LCS, because we believe to use LCS, because it is based on the phenomena of similarity rather than distance.[24]

4.5 Text Comparison Heuristics

Heuristic Algorithm is used to find the longest match of two hash strings by similarity method. In comparison at the document level scope, we compared two documents in accordance with common hashes and their fixed threshold. If the number of partitions in a subset of a larger crosses the threshold, then there is a possible similarity between the two documents. In this case, the comparison process is still at the paragraph level, is detected any similarity is shut down the operation. If the detection probability of similarity to the paragraph level, and then the process will continue on the wholesale level, otherwise the process terminates. In case of similarity between two sentences then, use longest common substring (LCS) to measured using the metric. Uncertainty the length of the longest corporate sequence is greater than the length multiplied by the minimum sentence threshold, then they determine similar chains in each of the strings, but this process will continue with the following sentence. We use a

heuristic algorithm of each level of the tree base on the document, paragraph and sentence.

A tree representation is created for each document to describe its logical structure. The root represents the document itself, the second level represents the paragraphs, and the leaf nodes contain the sentences. This representation is similar to the one used in CHECK [13]. It is intended to avoid unnecessary comparisons between several documents. Trees are then explored top-down and compared first at document level, then at paragraph level and finally at sentence level.

Heuristic algorithms for each level of the tree: Algorithm 1 (document level), Algorithm 2 (paragraph level), and Algorithm 3 (sentence level).At document level, two documents are compared according to their common hashes and a fixed threshold. If the number of hashes in the intersection subset is greater than the threshold, then there is a potential similarity between both documents. In that case, the comparison process continues at paragraph level, otherwise no similarity is detected and the process is stopped. If a possible similarity is detected at paragraph level, then the process continues at sentence level, otherwise the process terminates. If there is a possible similarity between two sentences, then it is measured using LCS metric. If the length of the longest common sequence is greater than the length of the minimum sentence multiplied by a threshold, then similar strings are identified in both sentences, otherwise the process continues with the next sentence.

Algorithm 1: Document level heuristic

Input : Doc1, Doc2 // Two input documents

Output: Matching similarity

Begin

DocMinSize = min (|Doc1|, |Doc2|)

DocIntersectionSize = |Doc1 ∩ Doc2|

If (DocIntersectionSize>= DocMinSize*DocThreshold)**Then**

//Possible similarity

//Check similarity at paragraph level

similarity = true

Else

similarity = false

End

Algorithm 2: Paragraph level heuristic

Input : Par1, Par1 // Two input paragraphs

Output: similarity

Begin

ParMinSize = min (|Par1|, |Par2|)

ParIntersectionSize = |Par1 ∩ Par2|

If (ParIntersectionSize>= ParMinSize*ParThreshold) **Then**

//Possible similarity

//Check similarity at sentence level

similarity = true

Else

similarity = false

End

Algorithm 3: Sentence level heuristic

Input : Sen1, Sen2

Output: similarity, similar substrings in Sen1 and Sen2

Begin

SenMinSize = min(|Sen1|, |Sen2|)

SenIntersectionSize = |Sen1 ∩ Sen2|

If (SenIntersectionSize>= SenMinSize*SenThreshold) **Then**

LongestCommonSeq = LCS (Sen1, Sen2)

If (|LongestCommonSeq| >= SenMinSize*SimilarityThreshold)**Then**

//Similarity detected

//Determine similar

//substrings

similarity = true

Else

similarity = false

Else

similarity = false

End

The precision , recall and F-Measure were used to evaluate detected as plagiarized statements regarding the total number of plagiarized statements at the document level on one hand, and to evaluate the retrieval process of detected documents as containing plagiarism regarding actual number of plagiarized documents in the corpora on the other hand. Performance results were measured using Recall (2) , Precision (3) and F-Measure (4)metrics.

$$\boldsymbol{Recall} = \frac{TP}{(TP+FN)} \quad (2)$$

$$\boldsymbol{Precision} = \frac{TP}{(TP+FP)} \quad (3)$$

$$\boldsymbol{F-Measure} = 2 * \frac{Recall*Precision}{Recall+Precision} \quad (4)$$

Where, true positives (TP): is the number of cases that plagiarized correctly detected. False positives (FP): is the number of cases that is detected False False negatives (FN): is the number of cases that plagiarized detected False.

4.6 Summary of the Framework

After addressed the problem of plagiarism detection in Arabic documents,where characteristics of Arabic language have been presented, and An operational framework

and detection method for Arabic Documents Plagiarism is introduced which, is go further for some hidden plagiarism such, as sentence structure change and synonym replacement. The main components of the framework is clearly described which, used heuristic algorithms for comparing fingerprints of Arabic documents at different logical levels (document, paragraph, and sentence) to pass up redundant comparisons.

CHAPTER V

DEVELOPMENT OF PLAGIARISM DETECTION TOOL FOR ARABIC DOCUMENTS

5.1 Introduction

In this chapter present the development of plagiarism detection tool for Arabic documents. The system of APDAM consist of two interface the first interface web-base build in PHP and MySql that allow create user from logion to the system. That system accept file upload and conversion after logion . If the file in that formation (.doc,.docx,html,.rtf,.dot) then it will converted to txt format (.txt) .the TXT file format using UTF-8 encoding.

5.2 Development Tool for Arabic Plagiarism Detection

5.2.1 NetBeans

NetBeans is an open-source integrated development environment (IDE) for developing with Java, PHP, C++, and other programming languages. NetBeans is also referred to as a platform of modular components used for developing Java desktop applications.

The Java Development Kit (JDK) is a software development environment used for developing Java applications and applets. It includes the Java Runtime Environment (JRE), an interpreter/loader (java), a compiler (javac), an archiver (jar), a documentation generator (javadoc) and other tools needed in Java development.

The JRE or the JDK. To run Java applications and applets, simply download the JRE. However, to develop Java applications and applets as well as run them, the JDK is needed.

5.2.2 XAMPP for MySQL Database:

XAMPP is a free and open source cross-platform web server solution stack package developed by Apache Friends [2]. consisting mainly of the Apache HTTP

Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages. As shows in figure 5.1 it is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing and deployment purposes. Everything needed to set up a web server – server application (Apache), database (MariaDB), and scripting language (PHP) – is included in an extractable file. XAMPP is also cross-platform, which means it works equally well on Linux, Mac and Windows. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server extremely easy as well.

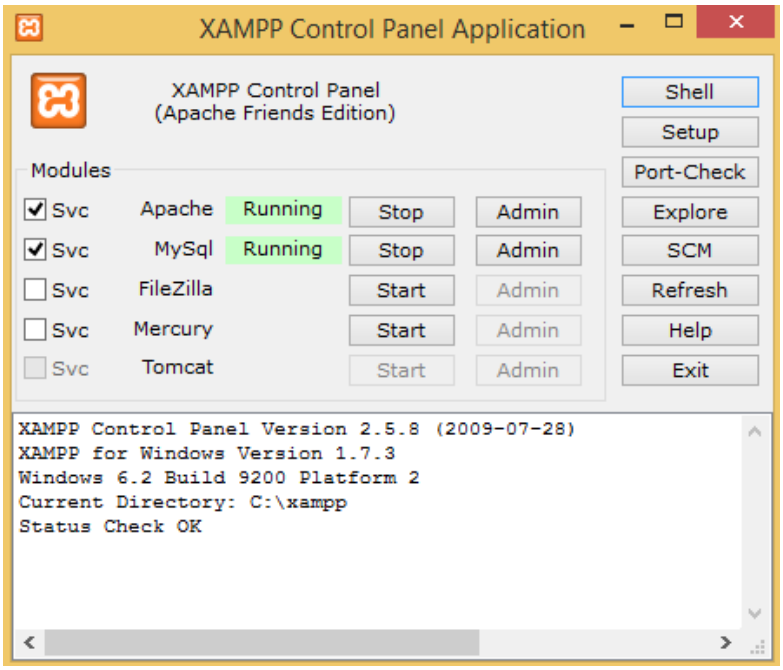


Figure 5.1 XAMPP Control Panel Application (Apache Friends Edition)

For Database we use MySQL Database because it is easy to handle the information and it useful to save the data when user lose his mobile , like saving cloud in server and we don't need more secure because the information in this application is generally it is not like Security Agencies. As whows in figure 5.2 present, the phpMyAdmin use to management MySQL database .

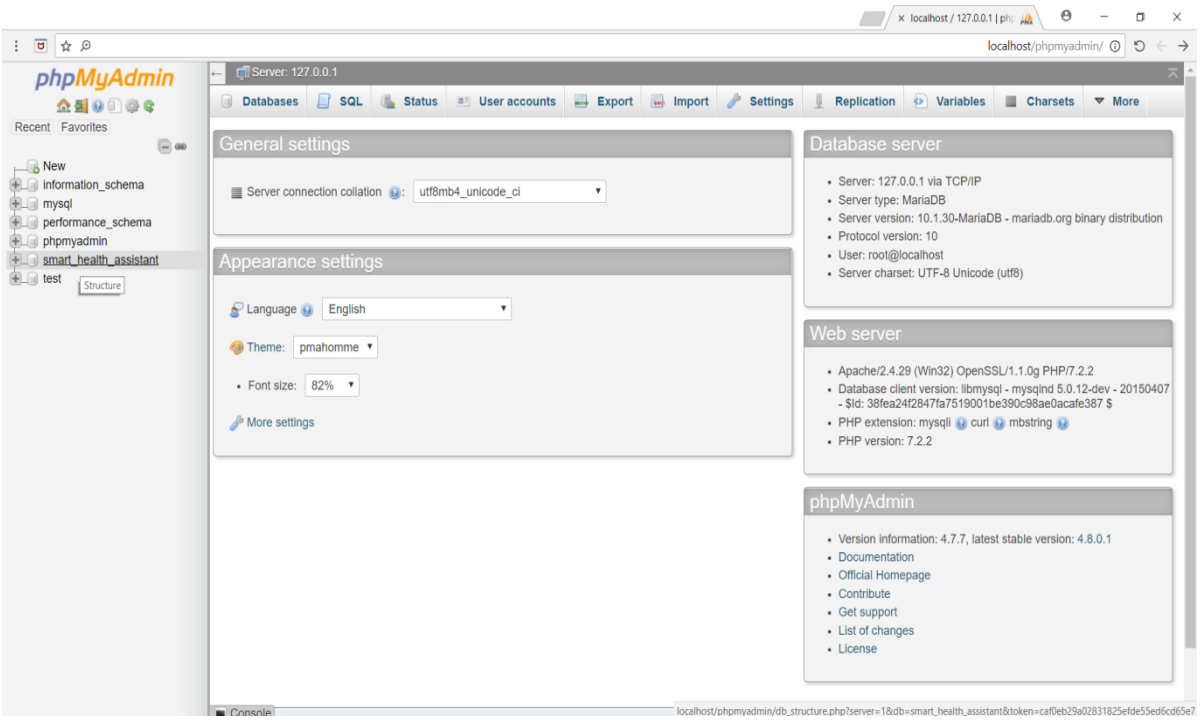


Figure 5.2: phpMyAdmin for Management Database

Were design some activities using the XML code as shows in figure 5.3 untike figure 5.5 and implement the activities using Java code for test basis. We also implement the MySQL database for our application. To connect with database (Papers), we use PHP code to insert the data into the database.

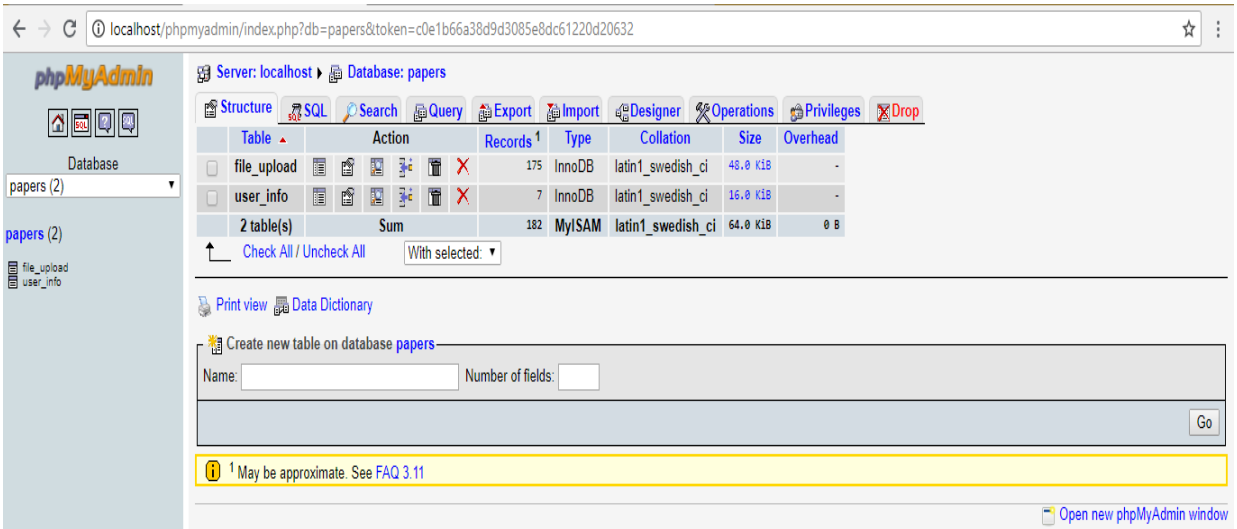


Figure 5.3: XAMPP for Papers Database

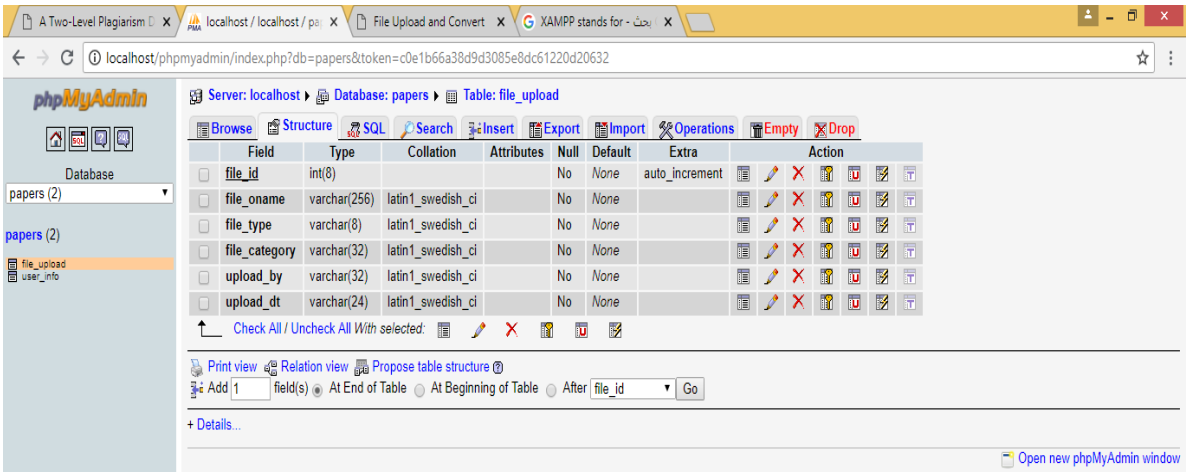


Figure 5.4: Arabic Document Uploaded in table users

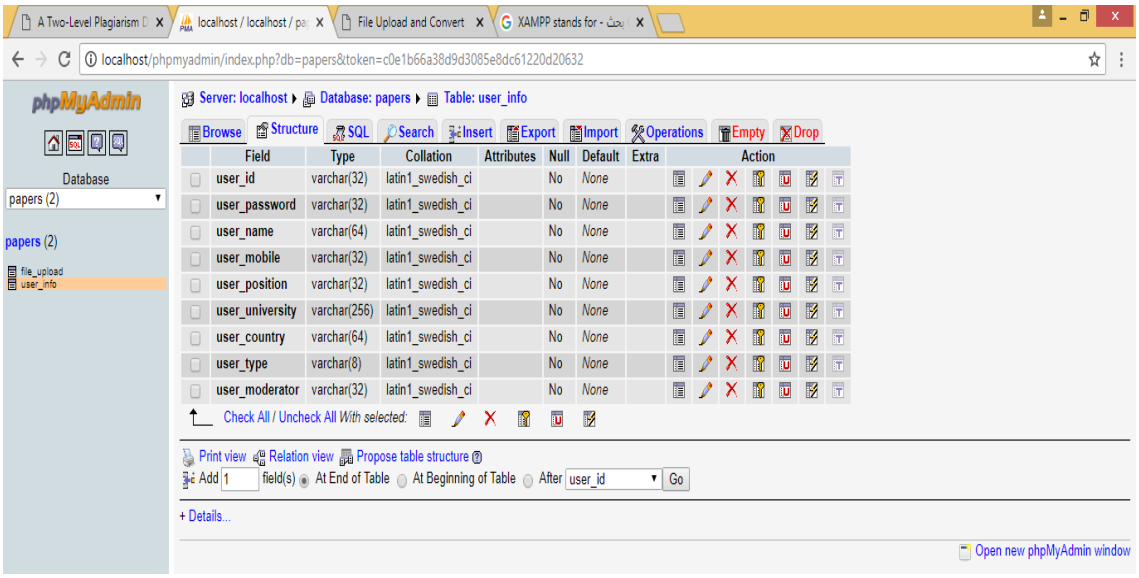


Figure 5.5 Arabic Document Uploaded in table File_upload

5.3 Development User Interface

The Arabic Plagiarism Detection System (WAPDS) web sited contents as showing in follows figuers. Firstly, registration on the website for uploads the Arabic document and then logion to system. It consist of four Menu bar Home Page, Upload File , Download , Modulator and Logout.

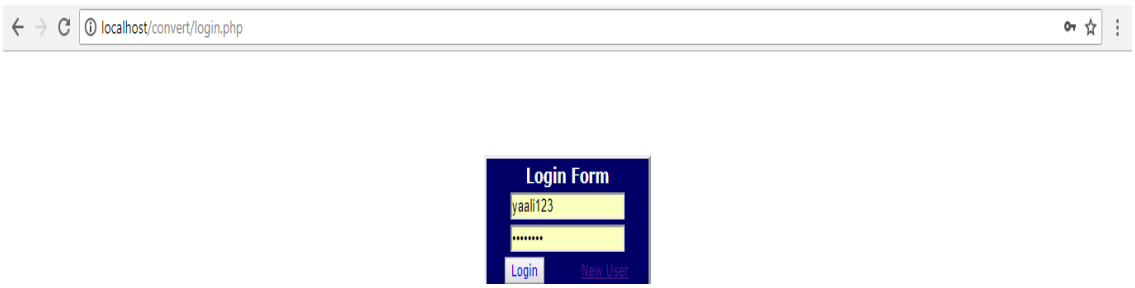


Figure 5.6: The website form Interface for Logion

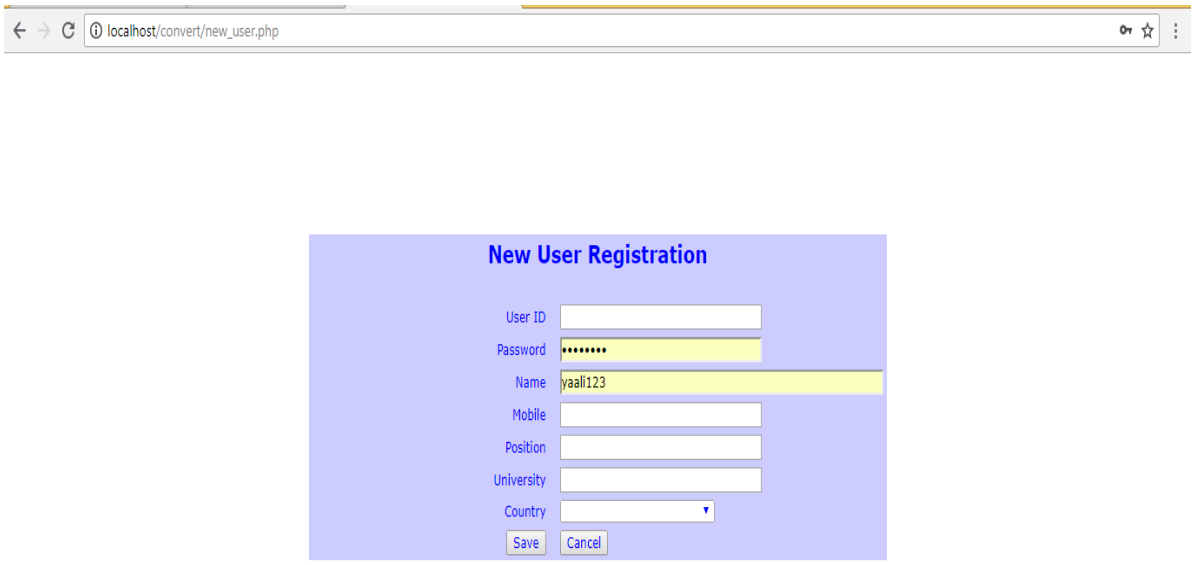


Figure 5.7: website new user registration

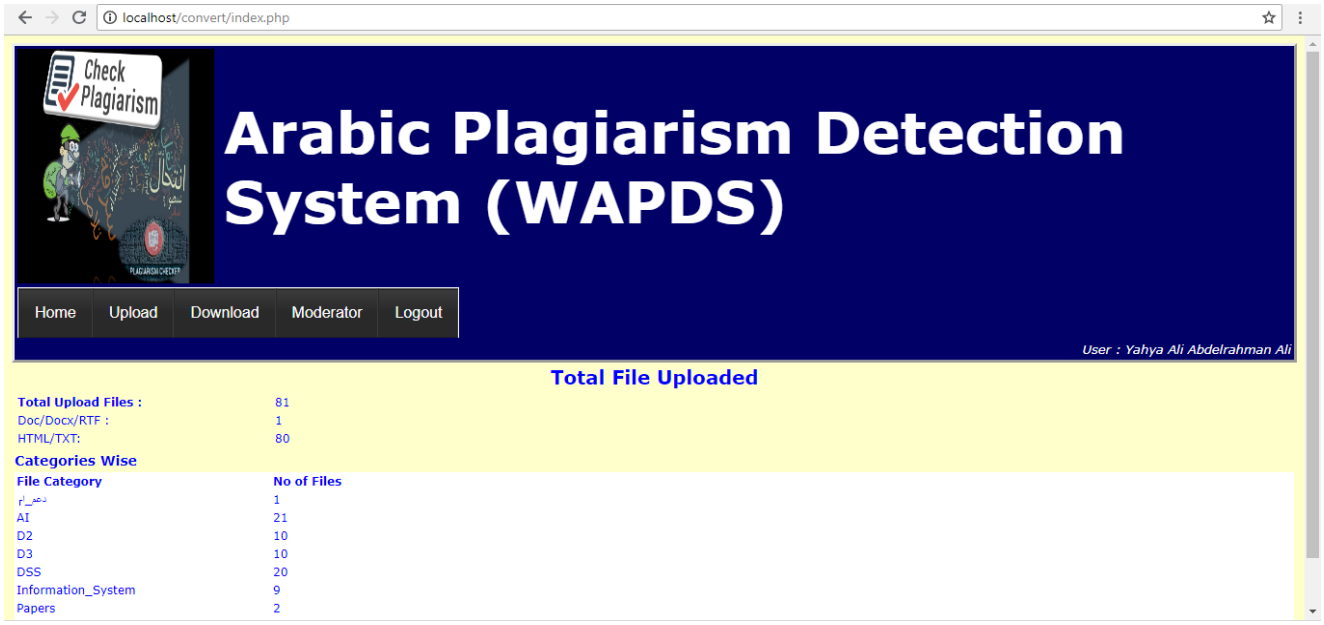


Figure 5.8: Interface Home Page

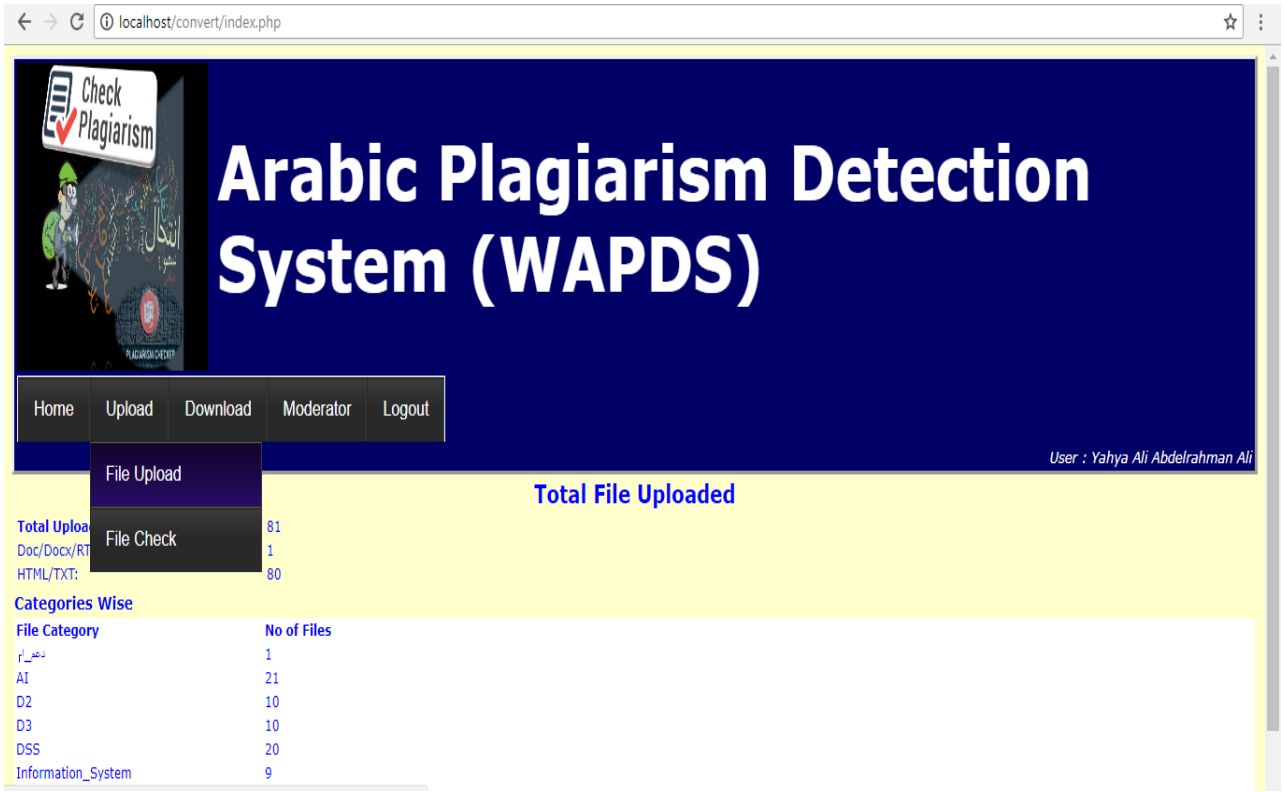


Figure 5.9: Interface upload menu page

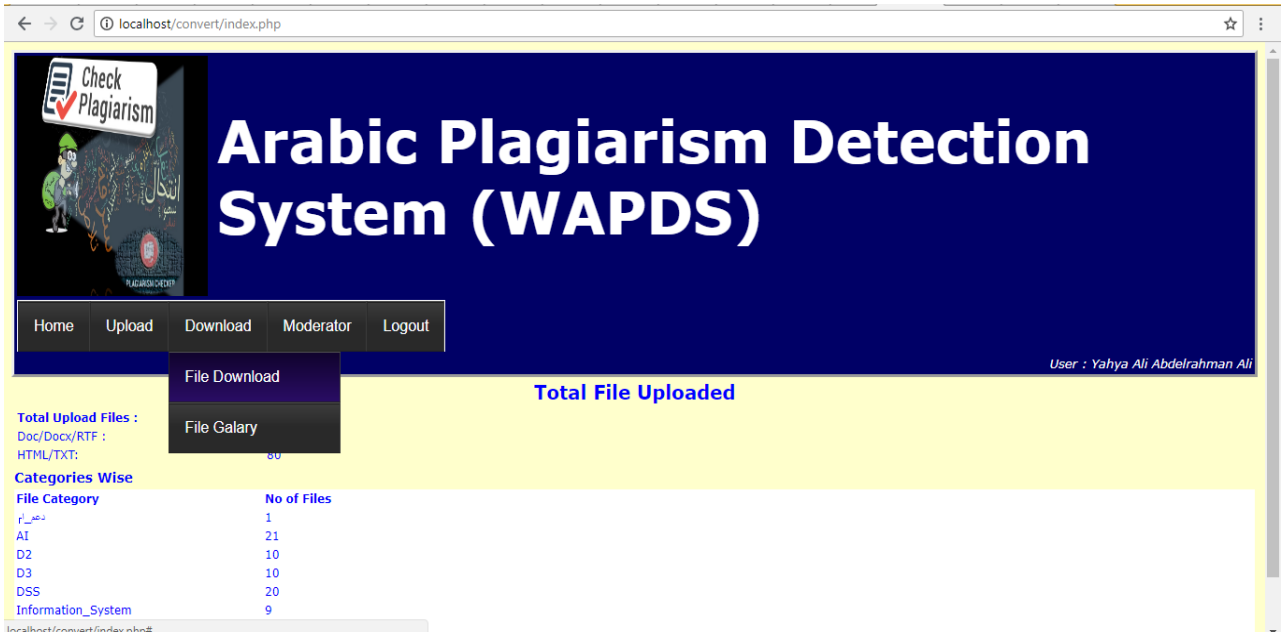


Figure 5.10 the download menu page

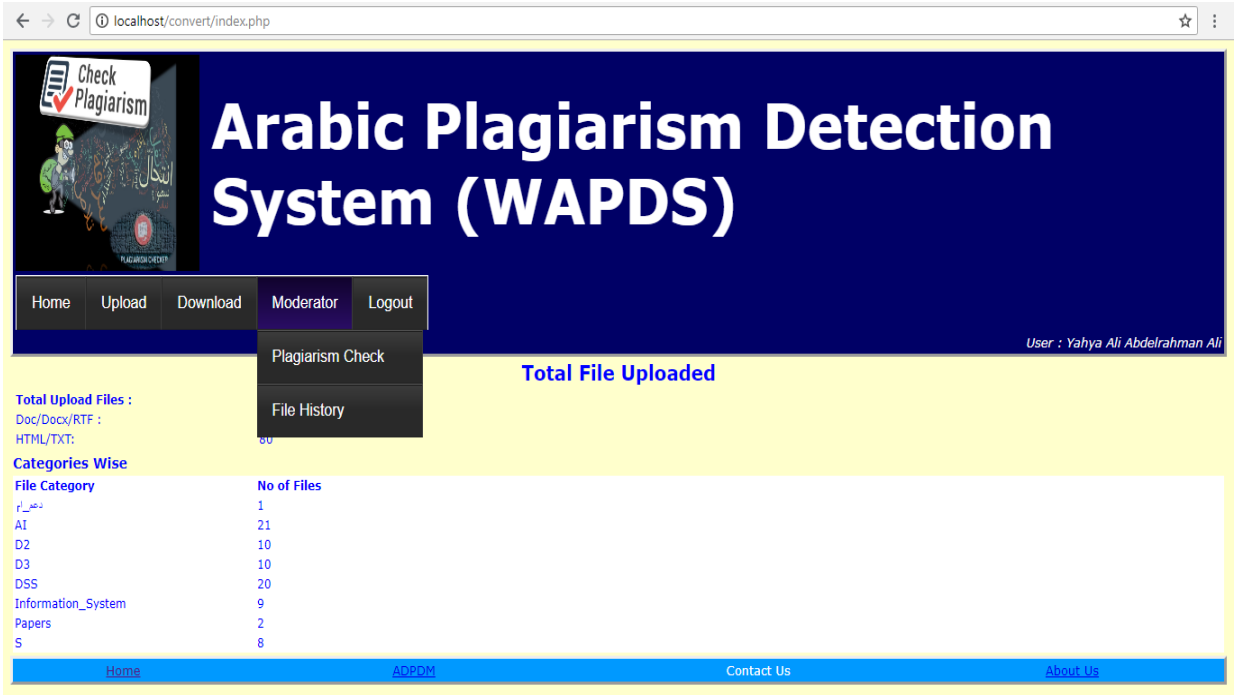


Figure 5.11 : The modulator task menu page

As showing on the follows, figures (5.12 to 5.18) are ADPDM Java Application using NetBeanse IDE 8.0.2 for plagiarism detection Arabic document tool.

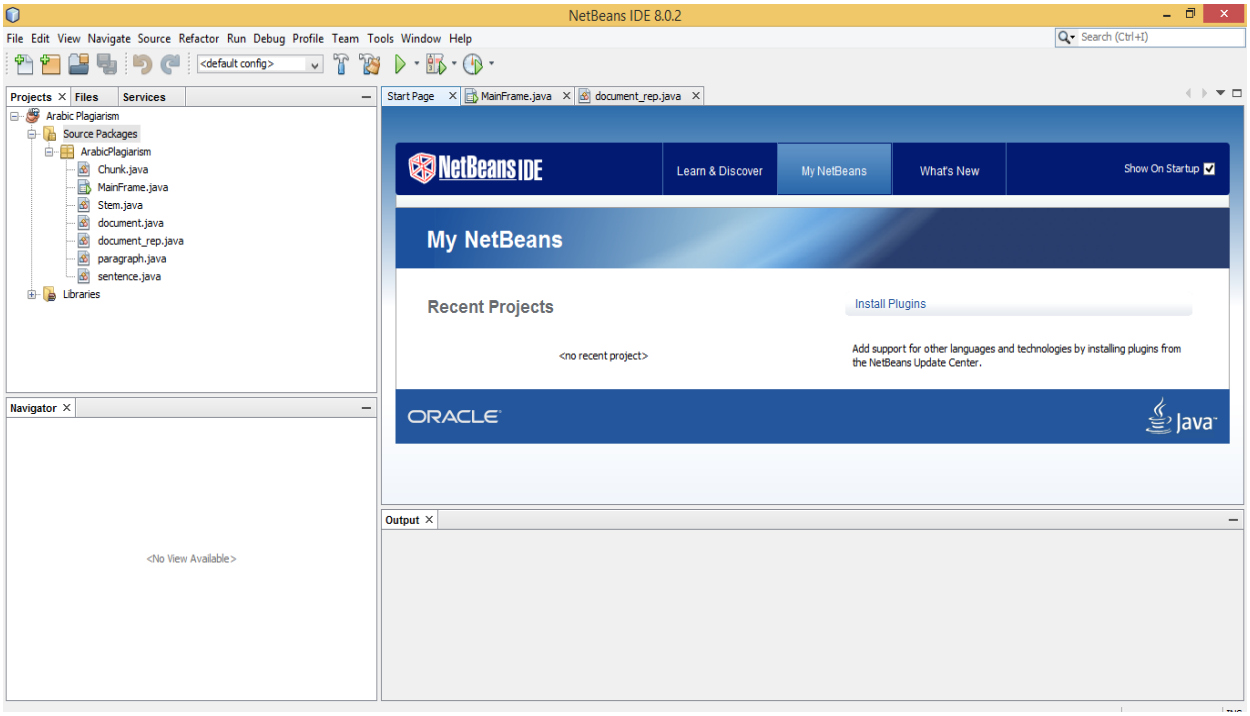


Figure5.12: ADPDM Java Application using NetBeanse IDE 8.0.2

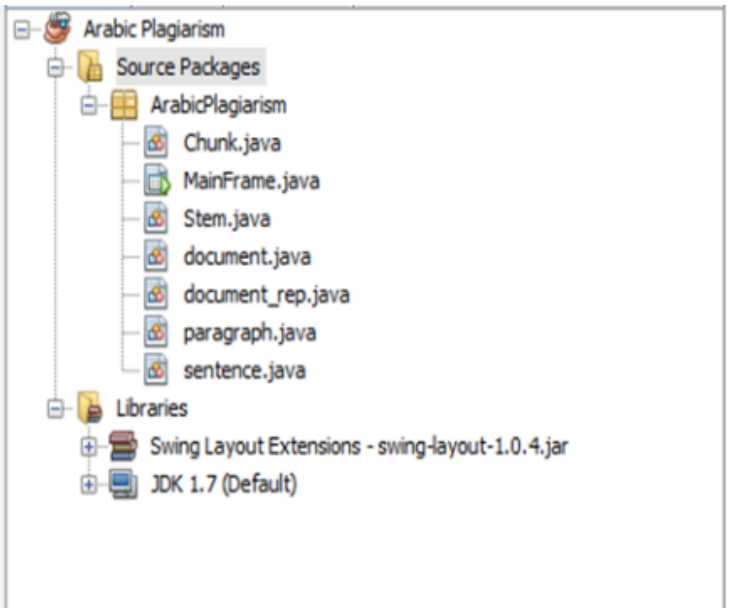


Figure5.13 : Source Packages Application using NetBeanse IDE 8.0.2 and Libraries

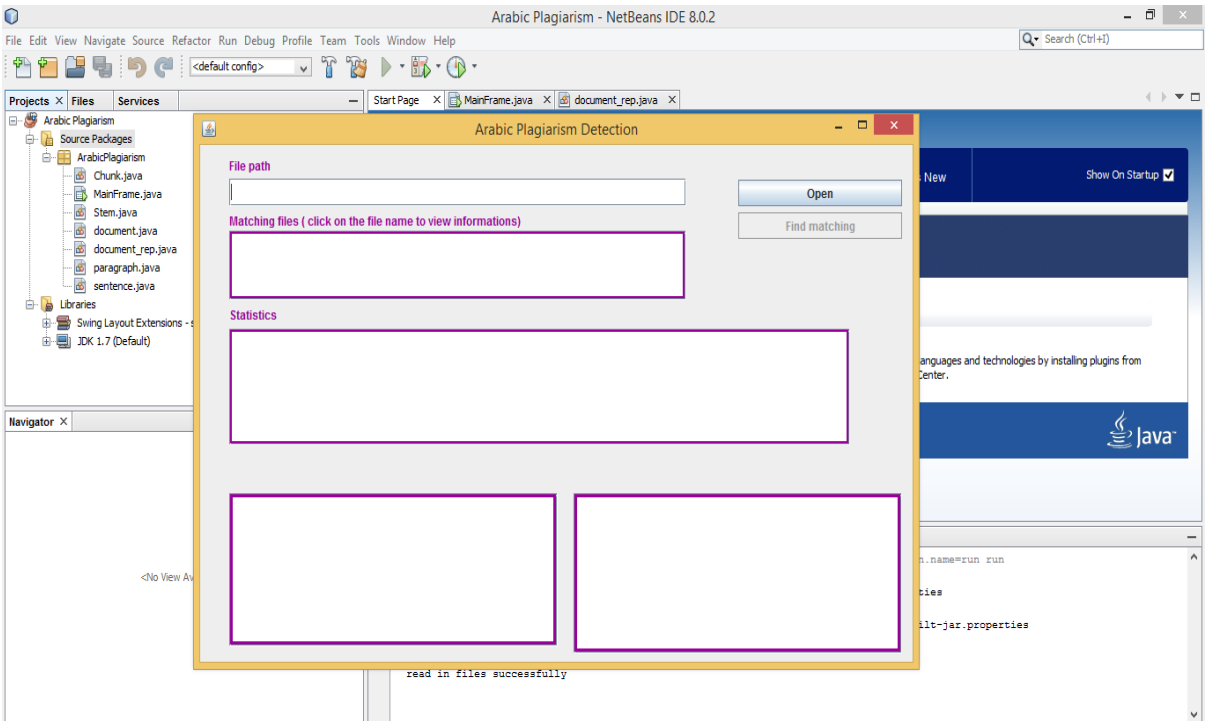


Figure 5.14: The ADPDM user interface

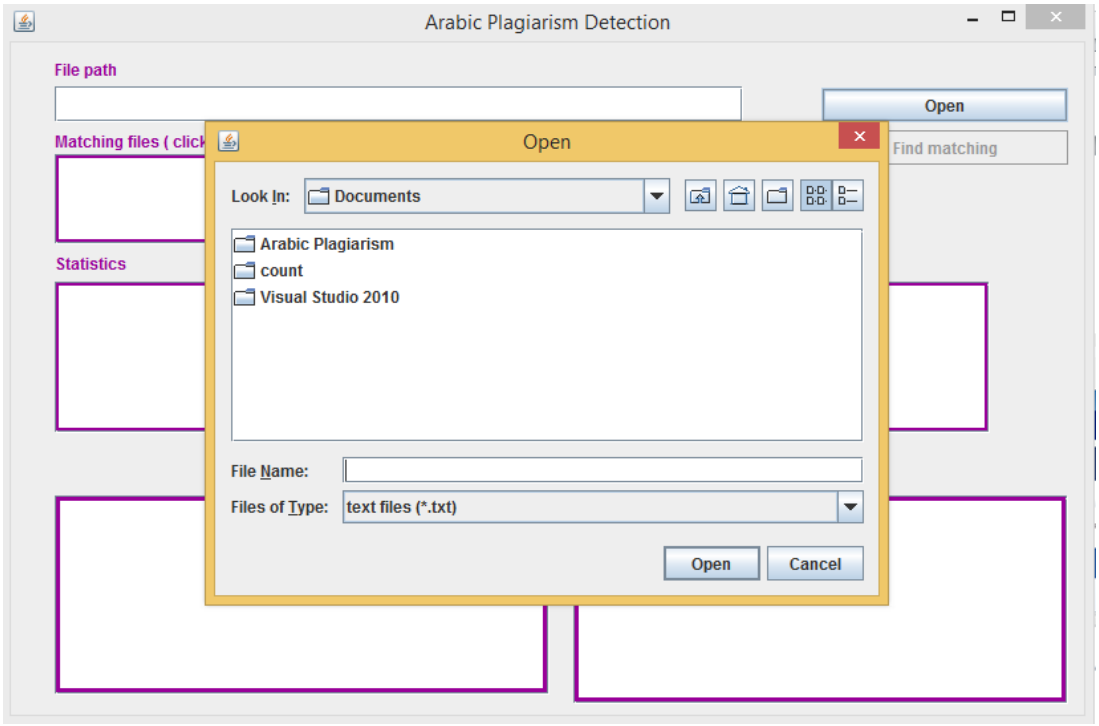


Figure5.15: Interface allow to show Overviews Arabic File (opensbutton)

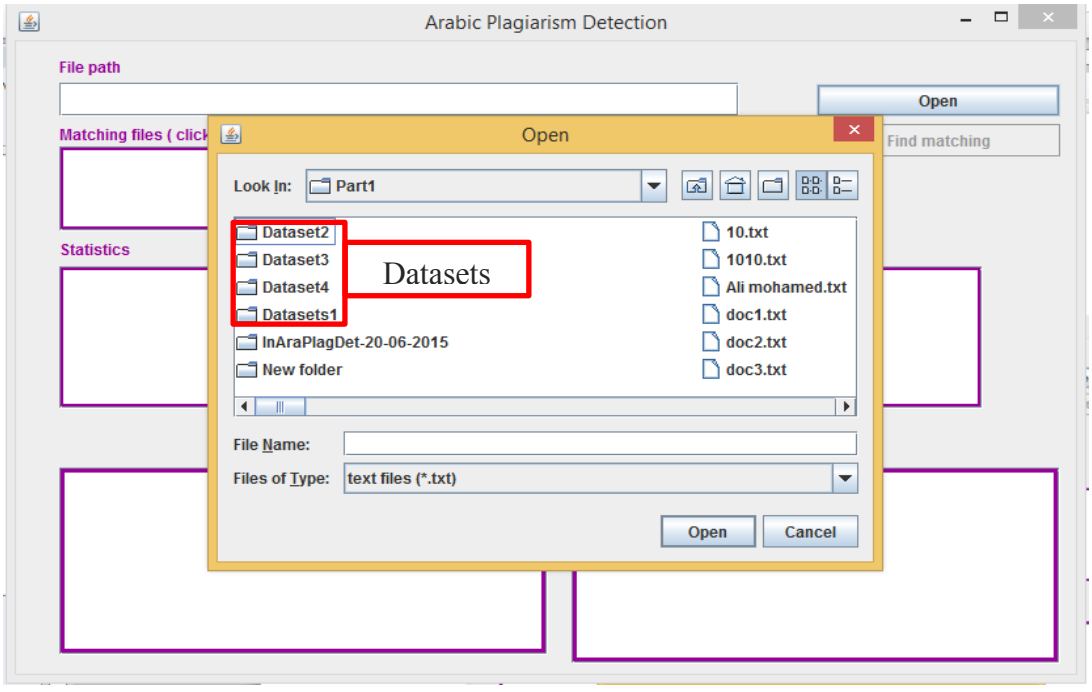


Figure 5.16: Interface allow to show Dataset selected to find the similarity

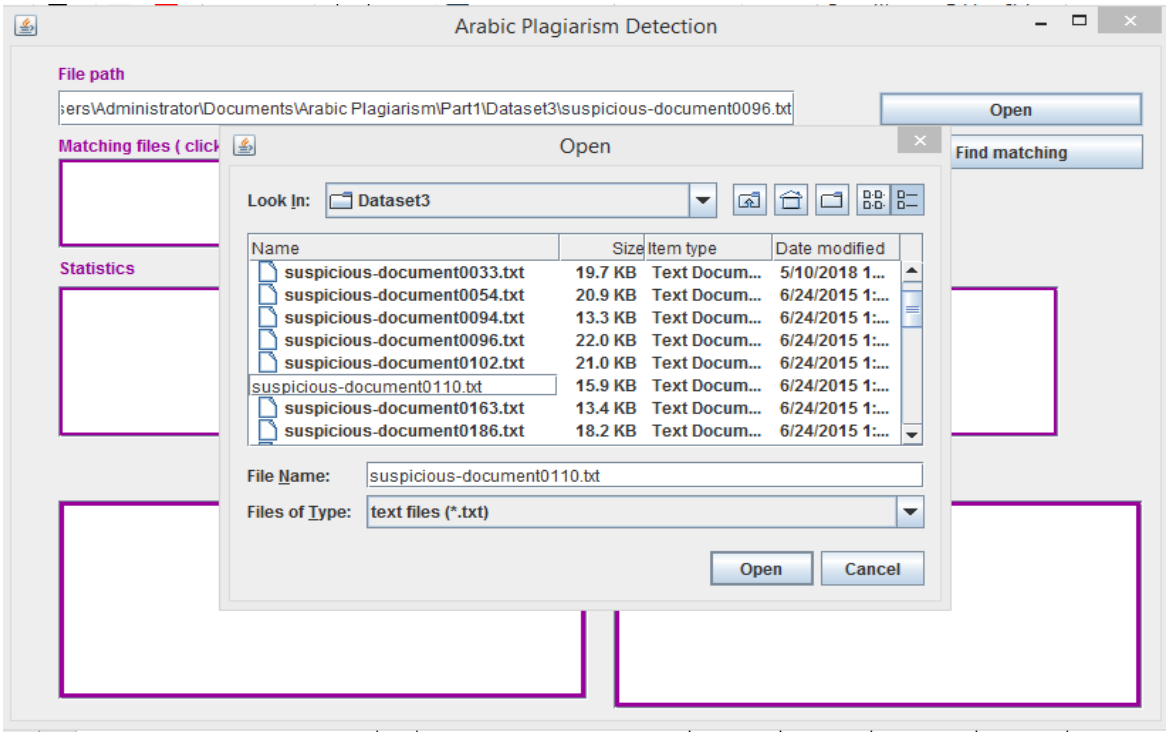


Figure 5.17: Interface allow select Dataset files in (.TXT) file format

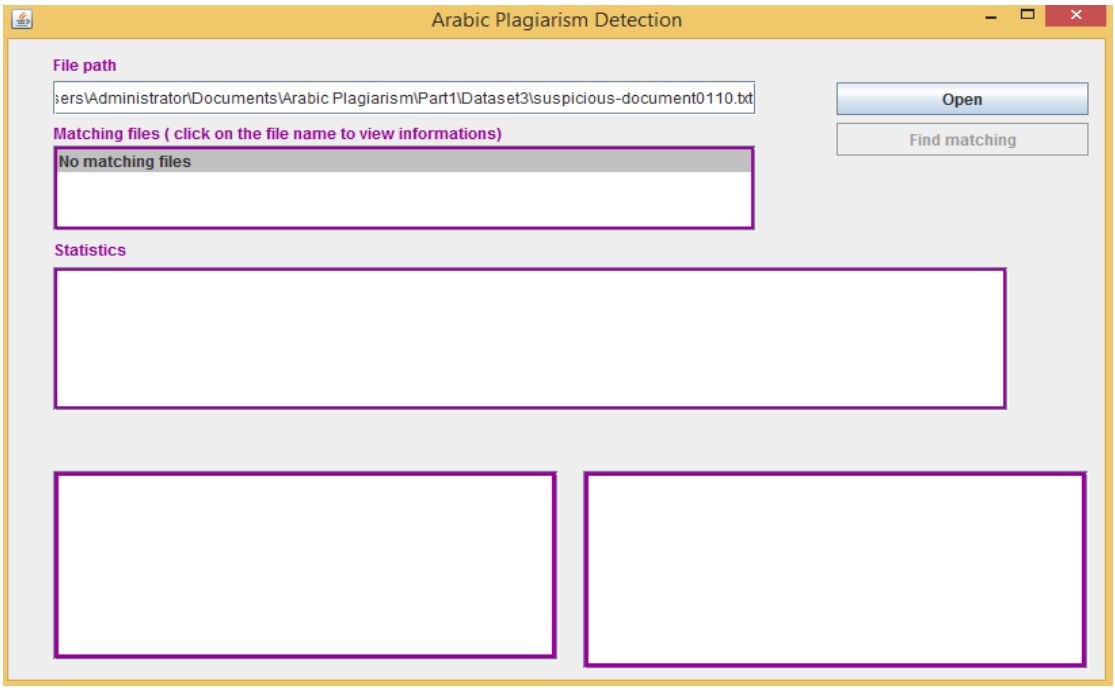


Figure 5.18 : Interface allow Shows matching files and statistical report

5.4 Summary

The summarization of this chapter is limited Development Tool for Arabic document Plagiarism Detection, that tool consist of website using PHP with XAMPP Control Panel Application for MySQL Database that allow for uploaded Arabic files in (.txt) format with UTF-8 encoded. When the file uploaded doesn't in (.txt) format con converted to that format, but must include (.docx , htm and rtf) for converted. Our main tool was built in java application NetBeans IDE 8.0.2 witch content of source Packages include(Chunk.java , document_rep.java, document.java , MainFrame.form , MainFrame.java ,paragraph.java , sentence.java ,Stem.java) and libraries. Finally we explained in details the development User Interface for website and ADPDM application.

CHAPTER VI

EXPERIMENTAL RESULT AND DISCUSSION

6.1 Introduction

Plagiarism detection process has four main stages shown in Figure 6 .1. The first stage is to submit a query document wherein we want to detect and judge plagiarism. Next includes pre-processing steps of the submitted document. Different techniques require different pre-processing steps as explained thoroughly in the methodology chapter.The third stage is to apply the plagiarism detection technique(s) to detect similar, probably plagiarised , patterns between the query document and the corpora. As a result, if plagiarism is found, plagiarized statements will be counted and highlighted, and a list of similar resources will be given.

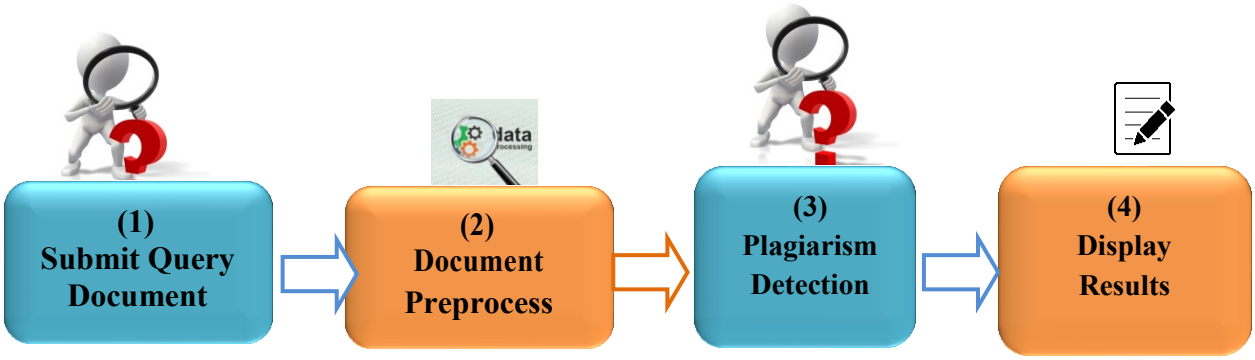


Figure 6.1: the Main step for ADPDM

This chapter discusses results of some experiments carried out using our plagiarism detection tools ADPDM and WCopyfind 4.1.5.exe. Then, we shed light on the preprocessing results from accomplishing stage 1 and 2 in Figure 6.1 This includes building the corpus collection, constructing the query documents, removing on essential data from both corpora and query documents to be ready for the last two stages. Next, we discuss the experimental results of fingerprints matching with and heuristic algorithm for each level with LCS matrix for plagiarism detection in Arabic documents that fulfill stage 3 and 4 in Figure 6.1. The completion of stage 4 designates the achievement of the goal of our study.

6.2 Experimental Evaluation

We implemented a prototype of Arabic plagiarism detection documents in Java and evaluated its performance on a handmade data test set of 102 Arabic documents of about 900 words each. We extracted tree type of data set each data set consist of 20 documents from different books available on AraPlagDet website [25]. We selected 3 datasets from the original documents and 1 dataset replaces the original documents randomly from 10%, 25%, and 40% 65%, 80% and 100% as follows.

Table 6.1: The datasets categories

Datasets	Number File	Size KB	No of word
Datasets1	30	182.84	19,141
Datasets2	30	297.48	31,175
Datasets3	30	535.61	56,693
Datasets4	12	60.48	5,685
Total	**102**	**1076.41**	**112694**

6.3 Datasets Information Details

6.3.1 Dataset1

As showing on table 6.2, they are 30 candidate documents were generated from each original document from AraPlagDet website the number of words in each document in range between 324 to 938 with size between 3.01 kb to 8.4 kb, the total of all dataset1 **19,141** words and total size **182.84kb.**

6.3.2 Dataset2

As showing on table 6.3, they are 30 candidate documents were generated from each original document from AraPlagDet website the number of words in each

document in range between 823 to 1782 with size between 7.01kb to 16.5kb ,the total of all dataset2 **31,175** words and total size **297.48 kb.**

6.3.3 Dataset3

As showing on table 6.4 , they are 30 candidate documents were generated from each original document from AraPlagDet website the number of words in each document in range between 1245 to 2540 words with size between 12.4 kb to 24.8kb ,the total of all dataset3 **56,693** words and total size **535.61kb.**

6.3.4 Dataset4 Structure change

As showing on table 6.5 candidate documents were generated from each original document Created from me from the book "أنظمة دعم القرار" and another document with same title that I mention it. the number of words in each document in range between 105 to 1387 words with size between 1.18 kb to 14.5kb ,the total of all dataset3 **5,685** words and total size **60.48kb.**

Table 6.2: The Arabic file Dataset1

Dataset1			
No	**File name**	**Size**	**No of Words**
1	suspicious-document0921.txt	3.21kb	324
2	suspicious-document0909.txt	3.56kb	362
3	suspicious-document0818.txt	4.57kb	482
4	suspicious-document0119.txt	5.40kb	547
5	suspicious-document0261.txt	5.40kb	582
6	suspicious-document0045.txt	5.41kb	527
7	suspicious-document0118.txt	5.49kb	577
8	suspicious-document0049.txt	5.69kb	552
9	suspicious-document0334.txt	5.74kb	597
10	suspicious-document1013.txt	5.83kb	547

11	suspicious-document0191.txt	5.85kb	584
12	suspicious-document0114.txt	5.85kb	582
13	suspicious-document0329.txt	5.90kb	662
14	suspicious-document0444.txt	5.97kb	643
15	suspicious-document0357.txt	5.99kb	650
16	suspicious-document0819.txt	6.06kb	548
17	suspicious-document0234.txt	6.07kb	662
18	suspicious-document0127.txt	6.08kb	639
19	suspicious-document0347.txt	6.08kb	637
20	suspicious-document0363.txt	6.09kb	658
21	suspicious-document0248.txt	6.7 kb	755
22	suspicious-document0743.txt	6.7 kb	705
23	suspicious-document0470.txt	6.8 kb	750
24	suspicious-document0485.txt	6.8 kb	705
25	suspicious-document0742.txt	7.2 kb	719
26	suspicious-document0580.txt	7.2 kb	762
27	suspicious-document0308.txt	7.2 kb	778
28	suspicious-document0729.txt	7.2 kb	765
29	suspicious-document0466.txt	8.4 kb	902
30	suspicious-document0507.txt	8.4 kb	938
Total Word Uploaded		**182.84kb**	**19,141**

Table 6.3: The Arabic file Dataset2

Dataset2

No	File name	Size	No of Words
1	suspicious-document0011.txt	7.01	823
2	suspicious-document0328.txt	7.03	735
3	suspicious-document0563.txt	7.34	801

4	suspicious-document0505.txt	7.25	761
5	suspicious-document0238.txt	7.45	749
6	suspicious-document0348.txt	7.53	782
7	suspicious-document0575.txt	7.74	860
8	suspicious-document0568.txt	7.95	860
9	suspicious-document0749.txt	7.95	790
10	suspicious-document0478.txt	7.99	860
11	suspicious-document0254.txt	8	873
12	suspicious-document0212.txt	8.04	798
13	suspicious-document0298.txt	8.14	932
14	suspicious-document0414.txt	8.21	917
15	suspicious-document0715.txt	8.41	850
16	suspicious-document0482.txt	9	986
17	suspicious-document0690.txt	9.22	978
18	suspicious-document0090.txt	10.01	1119
19	suspicious-document0981.txt	11.8	1062
20	suspicious-document0844.txt	12.01	1015
20	suspicious-document0844.txt	12.01	1015
21	suspicious-document0067.txt	11.7	1270
22	suspicious-document0630.txt	11.6	1191
23	suspicious-document0501.txt	11.7	1272
24	suspicious-document0642.txt	12.1	1280
25	suspicious-document0600.txt	12.1	1296
26	suspicious-document0093.txt	12.2	1263
27	suspicious-document0725.txt	13.8	1407
28	suspicious-document0184.txt	13.8	1348
29	suspicious-document0472.txt	13.9	1515
30	suspicious-document0549.txt	16.5	1782
	Total Word Uploaded	**297.48kb**	**31,175**

Table 6.4: The Arabic files Dataset3

Dataset 3			
No	**File name**	**Size**	**No of Words**
1	suspicious-document0708.txt	12.4kb	1245
2	suspicious-document0581.txt	12.5kb	1327
3	suspicious-document0785.txt	12.7kb	1348
4	suspicious-document0784.txt	13kb	1360
5	suspicious-document0255.txt	13.0kb	1428
6	suspicious-document0639.txt	13.01kb	1407
7	suspicious-document0094.txt	13.3kb	1375
8	suspicious-document0163.txt	13.4kb	1369
9	suspicious-document0825.txt	14kb	1558
10	suspicious-document0310.txt	14.9kb	1645
11	suspicious-document0021.txt	15.4kb	1662
12	suspicious-document0110.txt	15.9kb	1630
13	suspicious-document0311.txt	16kb	1721
14	suspicious-document0477.txt	16.9kb	1857
15	suspicious-document0219.txt	17kb	1706
16	suspicious-document0186.txt	18.2kb	1807
17	suspicious-document0302.txt	18.2kb	2052
18	suspicious-document0481.txt	19.3kb	2102
19	suspicious-document0033.txt	19.3kb	1854
20	suspicious-document0102.txt	21kb	2092
21	suspicious-document0841.txt	20.7	2359
22	suspicious-document0054.txt	21	2080
23	suspicious-document0446.txt	21	2280
24	suspicious-document0832.txt	22	2497
25	suspicious-document0383.txt	22	2358

26	suspicious-document0096.txt	22	2246
27	suspicious-document0447.txt	23.9	2617
28	suspicious-document0616.txt	24.1	2534
29	suspicious-document0655.txt	24.7	2637
30	suspicious-document0656.txt	24.8	2540
Total Word Uploaded		**535.61kb**	**56,693**

Table 6.5 : The Arabic file Dataset4

Dataset 4			
No	**File name**	**Size**	**No of Words**
2	دعم القرار١١.txt	1.42kb	135
3	دعم القرار١.txt	2.74kb	261
4	دعم القرار٣.txt	2.74kb	261
5	دعم القرار٢.txt	3.71kb	364
6	دعم القرار٤.txt	4.36kb	395
7	دعم القرار٧.txt	5.30kb	471
8	دعم القرار٩.txt	5.43kb	503
9	دعم القرار١٠.txt	5.66kb	529
10	دعم القرار٨.txt	5.76kb	564
11	دعم القرار٥.txt	7.68kb	710
12	دعم القرار٦.txt	14.5kb	1387
Total Word Uploaded		**60.48kb**	**5,685**

6.4 Results From our ADPDM Tools

We developed an **ADPDM** to compare two documents. This tool is simple and iterative that walks through files that already processed at the same time, table 6.6 untile table 6.12 and figure 6.2 untile figure 6.8 are visualizes results from our experiment.

After input 30 Arabic files, “suspicious-document” in (.TXT) format with different sizes between 3.21 to 8.4 KB and number of word in rang 823 up to 1171 words, the result we reached as showing in Table 6.6 and figure 6.2. 14% Proportion of plagiarism detection in dataset1.

Table 6.6: The result obtained by ADPDM on dataset1

Dataset1						
No	**File name**	**Size in KB**	**No of Words**	**File match**	**Total no of Word Detection**	**Time Duration in Second**
1	suspicious-document0921.txt	3.21	3	0	0	3.2
2	suspicious-document0909.txt	3.56	3	0	0	3.6
3	suspicious-document0818.txt	4.57	29.25	8	11	59.32
4	suspicious-document0119.txt	5.4	11.02	0	0	4
5	suspicious-document0261.txt	5.4	70.15	15	583	155.32
6	suspicious-document0045.txt	5.41	3	0	0	3.2
7	suspicious-document0118.txt	5.49	3	0	0	3.6
8	suspicious-document0049.txt	5.69	3	0	0	3.2
9	suspicious-document0334.txt	5.74	1654%	1	2	4.25
10	suspicious-document1013.txt	5.83	3.2	0	0	3.59
11	suspicious-document0191.txt	5.85	31.97	11	173	91.6
12	suspicious-document0114.txt	5.85	2.7	0	0	2.50
13	suspicious-document0329.txt	5.9	2.62	0	0	3.01
14	suspicious-document0444.txt	5.97	33.01	12	278	108.11
15	suspicious-document0357.txt	5.99	36.06	10	23	112.74
16	suspicious-document0819.txt	6.06	56.48	14	124	224.29
17	suspicious-document0234.txt	6.07	2	0	0	2.12
18	suspicious-document0127.txt	6.08	3	0	0	3.16
19	suspicious-document0347.txt	6.08	23.7	9	257	61
20	suspicious-document0363.txt	6.09	37.34	9	334	130.57

21	suspicious-document0248.txt	6.7	27.23	8	312	14.97
22	suspicious-document0743.txt	6.7	22.03	4	8	8.96
23	suspicious-document0470.txt	6.8	13.91	6	110	8.79
24	suspicious-document0485.txt	6.8	18.07	5	180	18.85
25	suspicious-document0742.txt	7.2	1.84	0	0	3.66
26	suspicious-document0580.txt	7.2	20.24	4	134	19.08
27	suspicious-document0308.txt	7.2	7.41	2	89	18.29
28	suspicious-document0729.txt	7.2	8.3	2	41	12.64
29	suspicious-document0466.txt	8.4	5.85	1	4	25
30	suspicious-document0507.txt	8.4	2.98	0	0	3.56
Total		**182.84**	**19141**	**121**	**2663**	**501**
					Percentage of All	**14%**

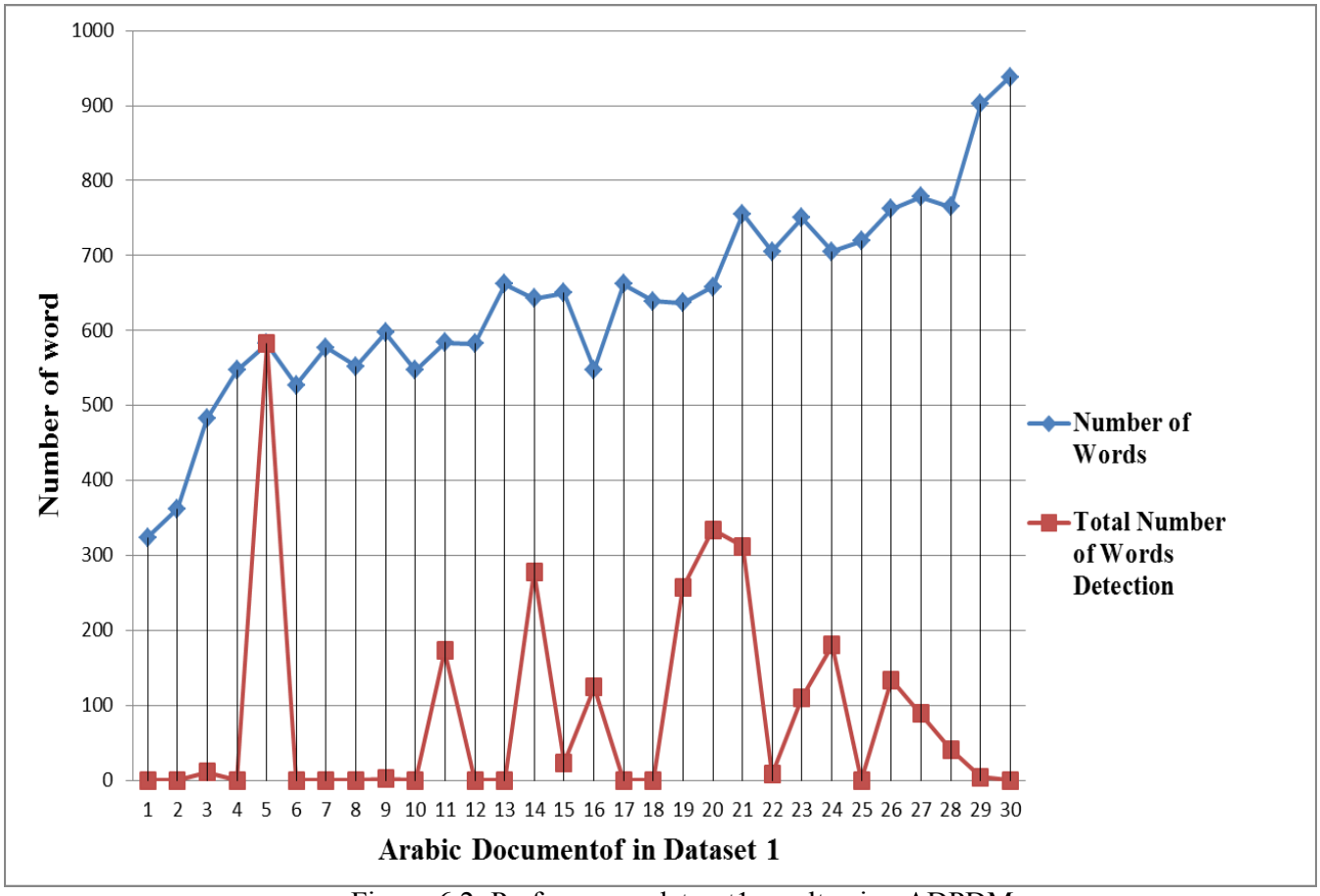


Figure 6.2: Performance dataset1 result using ADPDM

In dataset2 after input 30 Arabic files “suspicious-document” in (.TXT) format with different sizes between 7.01 to 16.4 KB and number of word in rang 823 up to 1171 words, the result we reached as showing in Table 6.7 and figure 6.3. 8.46% Proportion of plagiarism detection in dataset2.

Table 6.7: The result obtained by ADPDM on dataset2

Dataset2

No	File name	Size in KB	No of Words	File match	Total no of Word Detection	Time Duration in Second
1	suspicious-document0011.txt	7.01	823	7	7	2.8
2	suspicious-document0328.txt	7.03	735	15	177	2.89
3	suspicious-document0563.txt	7.34	801	15	177	29.25
4	suspicious-document0505.txt	7.25	761	0	0	11.02
5	suspicious-document0238.txt	7.45	749	0	0	70.15
6	suspicious-document0348.txt	7.53	782	15	160	2.9
7	suspicious-document0575.txt	7.74	860	14	49	2.93
8	suspicious-document0568.txt	7.95	860	14	412	3.09
9	suspicious-document0749.txt	7.95	790	0	0	16.54
10	suspicious-document0478.txt	7.99	860	12	88	3.2
11	suspicious-document0254.txt	8	873	10	465	31.97
12	suspicious-document0212.txt	8.04	798	4	8	2.7
13	suspicious-document0298.txt	8.14	932	10	486	2.62
14	suspicious-document0414.txt	8.21	917	8	131	33.01
15	suspicious-document0715.txt	8.41	850	0	0	36.06
16	suspicious-document0482.txt	9	986	6	138	56.48
17	suspicious-document0690.txt	9.22	978	8	83	2.13
18	suspicious-document0090.txt	10.01	1119	3	4	2.6
19	suspicious-document0981.txt	11.8	1062	4	64	23.7
20	suspicious-document0844.txt	12.01	1015	2	2	37.34
21	suspicious-document0067.txt	11.7	1270	0	0	27.23
22	suspicious-document0630.txt	11.6	1191	0	0	22.03

23	suspicious-document0501.txt	11.7	1272	1	1	13.91
24	suspicious-document0642.txt	12.1	1280	1	11	18.07
25	suspicious-document0600.txt	12.1	1296	2	11	1.84
26	suspicious-document0093.txt	12.2	1263	0	0	20.24
27	suspicious-document0725.txt	13.8	1407	1	4	7.41
28	suspicious-document0184.txt	13.8	1348	3	7	8.3
29	suspicious-document0472.txt	13.9	1515	5	28	5.85
30	suspicious-document0549.txt	16.5	1782	6	135	2.98
	Total	**297.48**	**31,175**	**166**	**2648**	**501**
					Percentage of All	**8.46%**

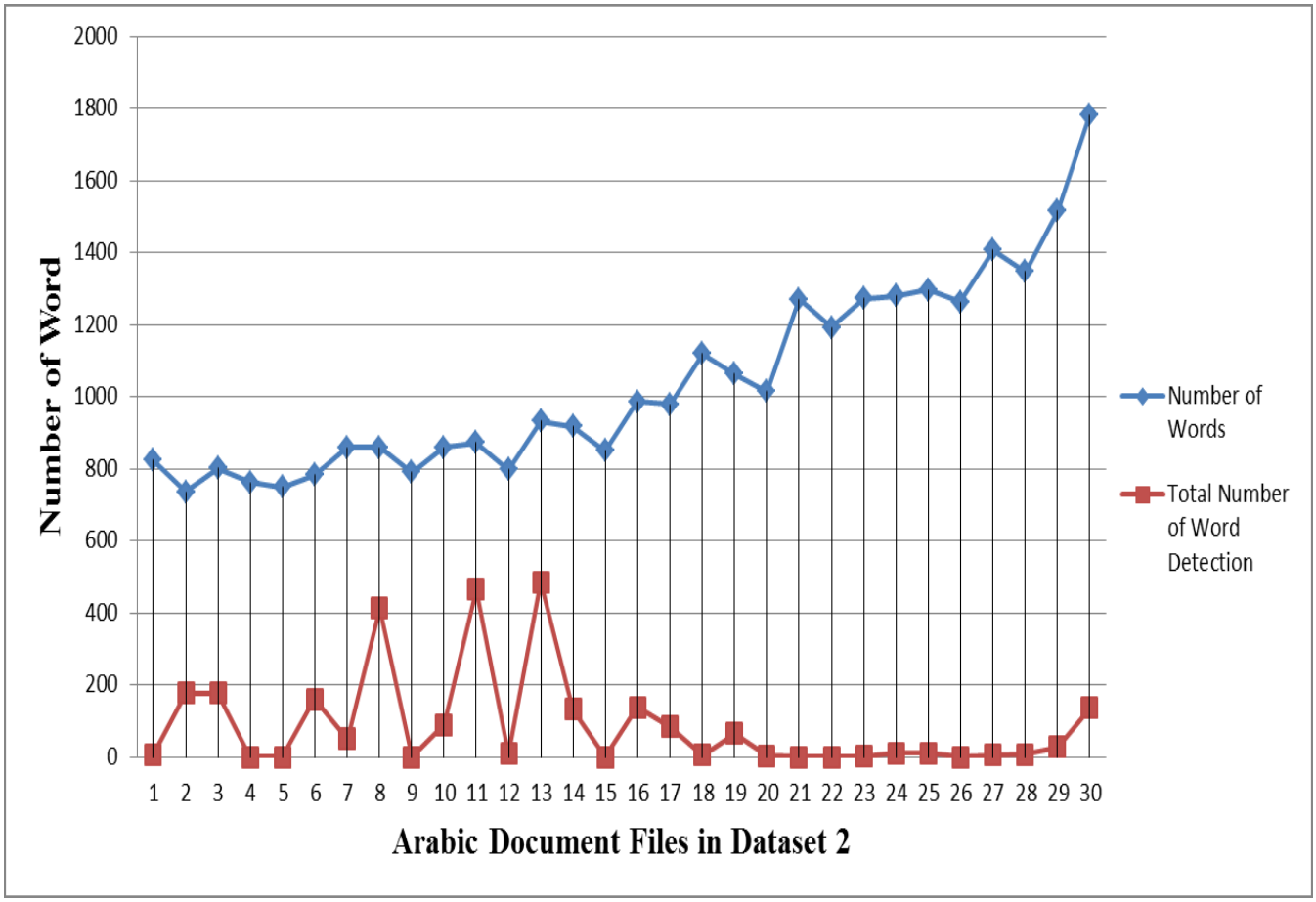


Figure 6.3: ADPDM result performance on Dataset2

As showing in Table 6.8 and figure 6.1 dataset3 after input 30 Arabic files "suspicious-document" in (.TXT) format with different sizes between 12.4 to 24.8 KB

and number of word in rang 1245 up to 2540 words, the result we reached. 18% Proportion of plagiarism detection in dataset3.

Table 6.8: The result obtained by ADPDM on dataset3

Dataset3

No	File name	Size in KB	No of Words	File match	Total no of Word Detection	Time Duration in Second
1	suspicious-document0708.txt	12.4	1245	8	60	63.76
2	suspicious-document0581.txt	12.5	1327	12	279	204.34
3	suspicious-document0785.txt	12.7	1348	0	0	5.25
4	suspicious-document0784.txt	13	1360	0	0	4.94
5	suspicious-document0255.txt	13	1428	3	10	177.14
6	suspicious-document0639.txt	13.01	1407	15	140	162.59
7	suspicious-document0094.txt	13.3	1375	0	0	2
8	suspicious-document0163.txt	13.4	1369	1	9	120
9	suspicious-document0825.txt	14	1558	1	200	71.65
10	suspicious-document0310.txt	14.9	1645	8	827	250.98
11	suspicious-document0021.txt	15.4	1662	0	0	120.23
12	suspicious-document0110.txt	15.9	1630	0	0	0
13	suspicious-document0311.txt	16	1721	6	3276	180
14	suspicious-document0477.txt	16.9	1857	8	1575	100
15	suspicious-document0219.txt	17	1706	15	1300	0
16	suspicious-document0186.txt	18.2	1807	11	51	288.3
17	suspicious-document0302.txt	18.2	2052	11	568	170
18	suspicious-document0481.txt	19.3	2102	10	256	206.33
19	suspicious-document0033.txt	19.3	1854	10	243	4.6
20	suspicious-document0102.txt	21	2092	0	0	2.53
21	suspicious-document0841.txt	20.7	2359	9	71	105.27
22	suspicious-document0054.txt	21	2080	7	68	111.68
23	suspicious-document0446.txt	21	2280	6	299	71.61
24	suspicious-document0832.txt	22	2497	6	15	56.53

25	suspicious-document0383.txt	22	2358	5	523	117.62
26	suspicious-document0096.txt	22	2246	4	25	153.83
27	suspicious-document0447.txt	23.9	2617	3	81	54.97
28	suspicious-document0616.txt	24.1	2534	2	38	27.41
29	suspicious-document0655.txt	24.7	2637	1	22	30.84
30	suspicious-document0656.txt	24.8	2540	0	0	14.93
	Total	**535.61**	**56693**	**100**	**9936**	**1696.45**
					Percentage of All	**18%**

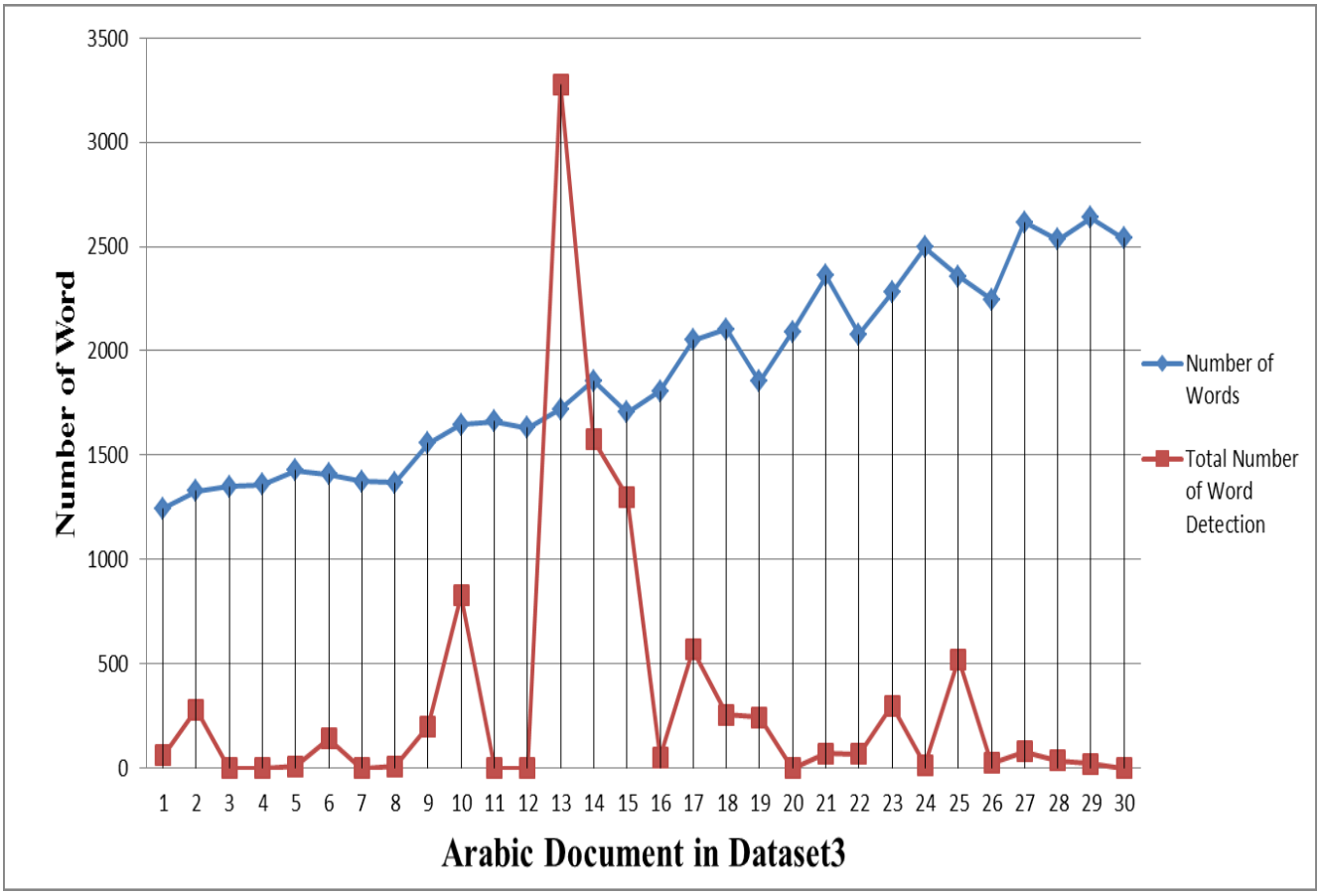


Figure 6.4: Performance Dataset3 result using ADPDM

As showing in Table 6.9 and figure 6.5 dataset3 with contain 30 Arabic files "suspicious-document" in (.TXT) format with different sizes between 1.18 to 14.5 KB and number of word in rang 105 up to 1357 words, the result we reached. 94% Proportion of plagiarism detection in dataset4.

Table 6.9: The result obtained by ADPDM on dataset4

Dataset4

No	File name	Size	No of Words	File match	File match Name	Total no of Word Detection	Time Duration
1	txt.١٢ دعم القرار	1.18kb	105	7	6,5,4, 3,2,1,11	351	36.22
2	txt.١١ دعم القرار	1.42kb	135	8	9,8,6,5,4,1,2	655	56.85
3	txt.١ دعم القرار	2.74kb	261	7	9,8,6,5,4,2	989	79.32
4	txt.٣ دعم القرار	2.74kb	261	6	9,8,6,5,4,2	731	61.88
5	txt.٢ دعم القرار	3.71kb	364	5	9,8,6,5,4	679	71.42
6	txt.٤ دعم القرار	4.36kb	395	4	9,8,6,5	480	86.57
7	txt.٧ دعم القرار	5.30kb	471	0	-	0	3
8	txt.٩ دعم القرار	5.43kb	503	3	8,6,5	598	229.13
9	txt.١٠ دعم القرار	5.66kb	529	0	-	0	2
10	txt.٨ دعم القرار	5.76kb	564	2	5,6	812	49.55
11	txt.٥ دعم القرار	7.68kb	710	1	6	44	6.85
12	txt.٦ دعم القرار	14.5kb	1387	0	-	0	2
		Total Word Uploaded	**5685**	**43**	**Total Word Detection**	**5339**	**Word Range**
					Percentage of All	**94%**	**105 - 1387**

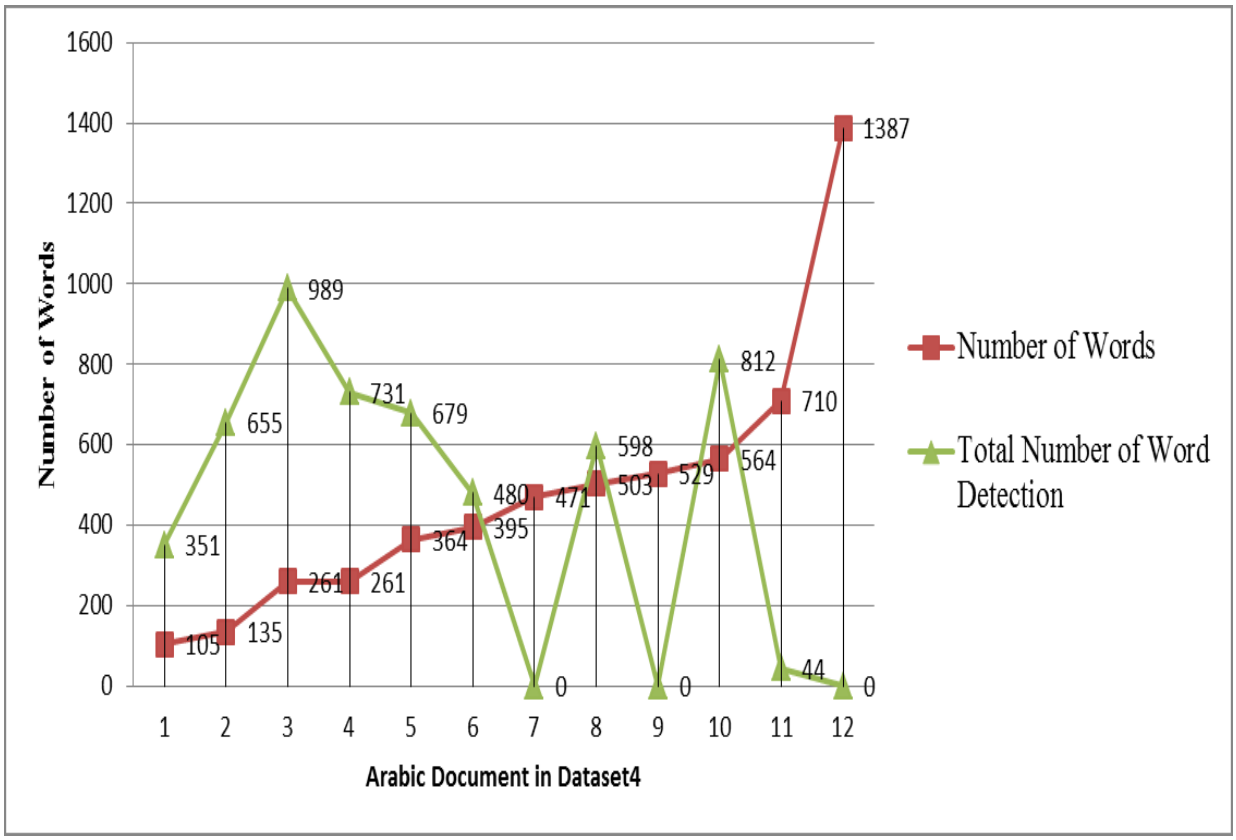


Figure 6.5 :Performance Dataset4 result using ADPDM

6.5 Results From WCopyfind64.4.1.5 Tools

In figure 6.6 and figure 6.7 as showing below present 30 Arabic files was uploaded to find the plagiarism. An experimental dataset1 tested by Wcopyfind4.1.5 application, 0 files plagiarized found with total percentage 0%.

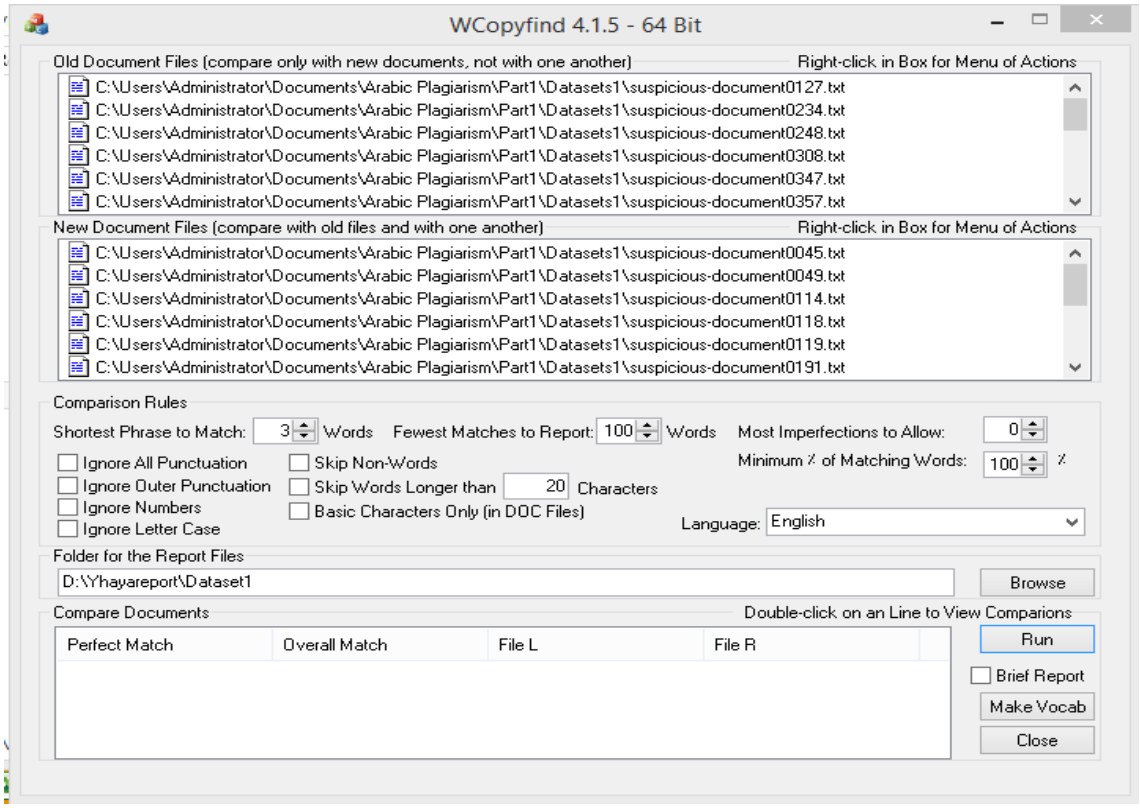


Figure 6.6 :Wcopyfind 4.1.5 uploaded Arabic files Dataset1

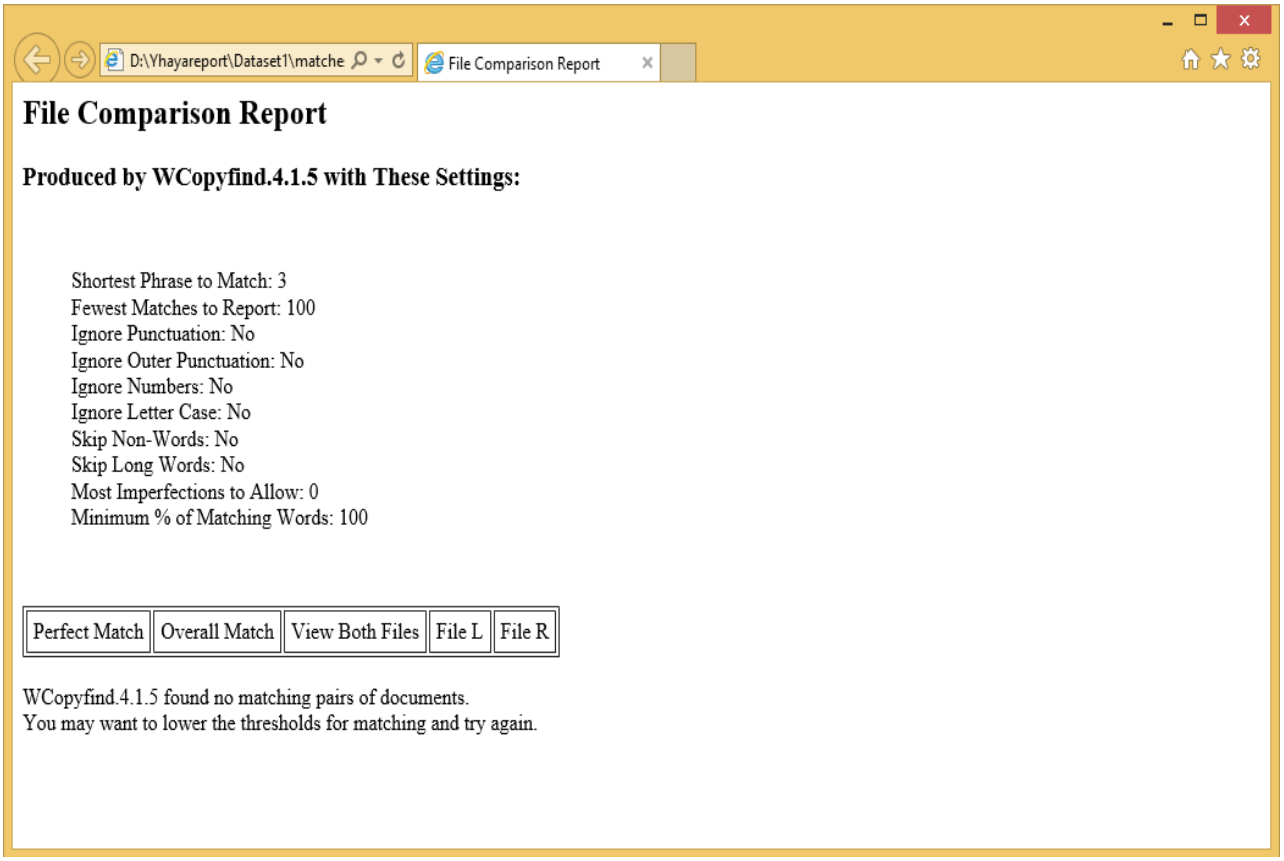


Figure 6.7 :Wcopyfind4.1.5 Report Arabic files Dataset1

In figure 6.8 and figure 6.9 as showing below present 30 Arabic files was uploaded to find the plagiarism. An experimental dataset2 tested by Wcopyfind4.1.5 application, 0 files plagiarized found with total percentage 0%.

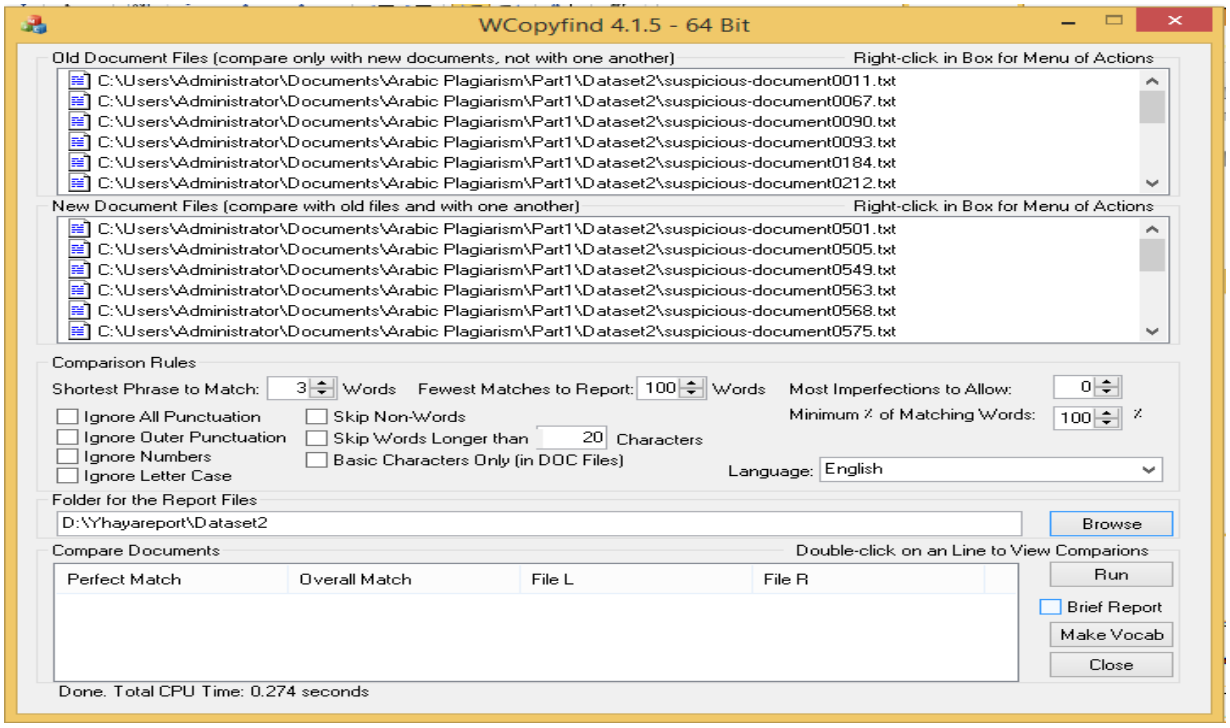


Figure 6.8 : Wcopyfind 4.1.5 uploaded Arabic files for Dataset2

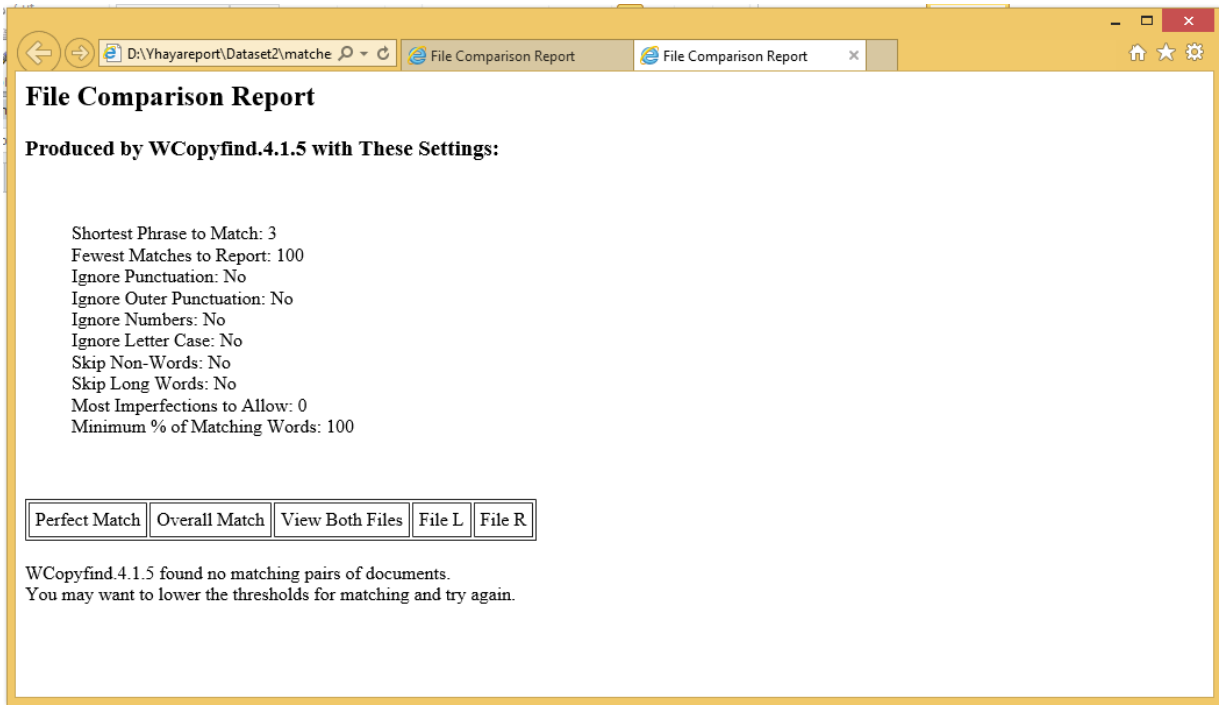


Figure 6.9 :Wcopyfind 4.1.5 Report for Dataset2

In figure 6.10 and figure 6.11 as showing below present 30 Arabic files was uploaded to find the plagiarism. An experimental dataset3 tested by Wcopyfind4.1.5 application, 7 files plagiarized found with total percentage 6.33%.

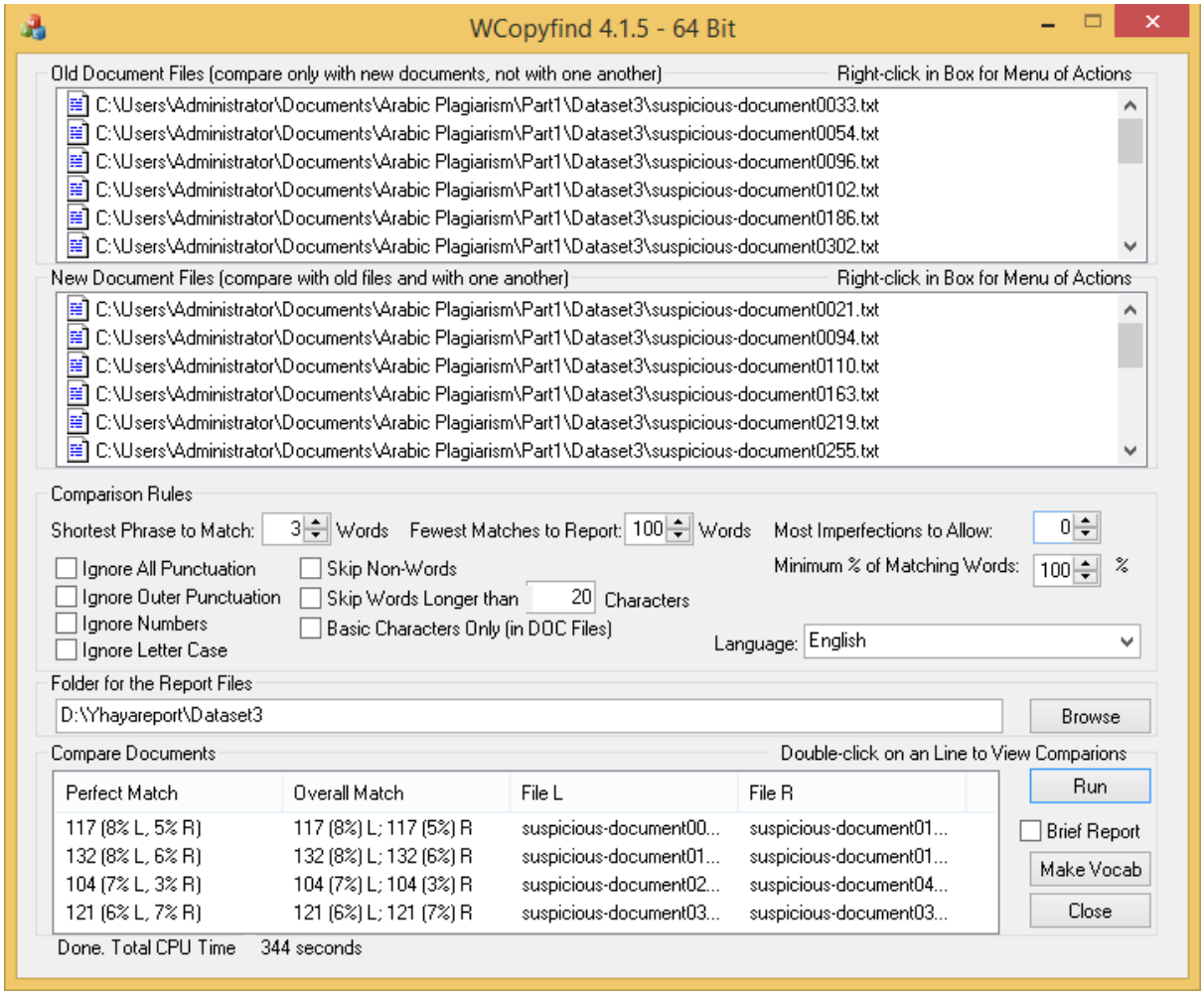


Figure 6.10 : Wcopyfind 4.1.5 Dataset3 Arabic files uploaded

D:\Yhayareport\Dataset3\matches.html | File Comparison Report | File Comparison Report

File Comparison Report

Produced by WCopyfind.4.1.5 with These Settings:

Shortest Phrase to Match: 3
Fewest Matches to Report: 100
Ignore Punctuation: No
Ignore Outer Punctuation: No
Ignore Numbers: No
Ignore Letter Case: No
Skip Non-Words: No
Skip Long Words: No
Most Imperfections to Allow: 0
Minimum % of Matching Words: 100

Perfect Match	Overall Match	View Both Files	File L	File R
117 (8% L, 5% R)	117 (8%) L; 117 (5%) R	Side-by-Side	suspicious-document0094.txt	suspicious-document0102.txt
132 (8% L, 6% R)	132 (8%) L; 132 (6%) R	Side-by-Side	suspicious-document0110.txt	suspicious-document0102.txt
104 (7% L, 3% R)	104 (7%) L; 104 (3%) R	Side-by-Side	suspicious-document0255.txt	suspicious-document0447.txt
121 (6% L, 7% R)	121 (6%) L; 121 (7%) R	Side-by-Side	suspicious-document0311.txt	suspicious-document0310.txt

WCopyfind.4.1.5 found 4 matching pairs of documents.

Figure 6.11: Wcopyfind 4.1.5 Plagiarised detection Report for Dataset3

In figure 6.12, figure 6.13 and figure 6.14 as showing below present 12 Arabic files was uploaded to find the plagiarism. An experimental dataset4 tested by Wcopyfind4.1.5 application, 7 files plagiarized found with total percentage 84.33%.

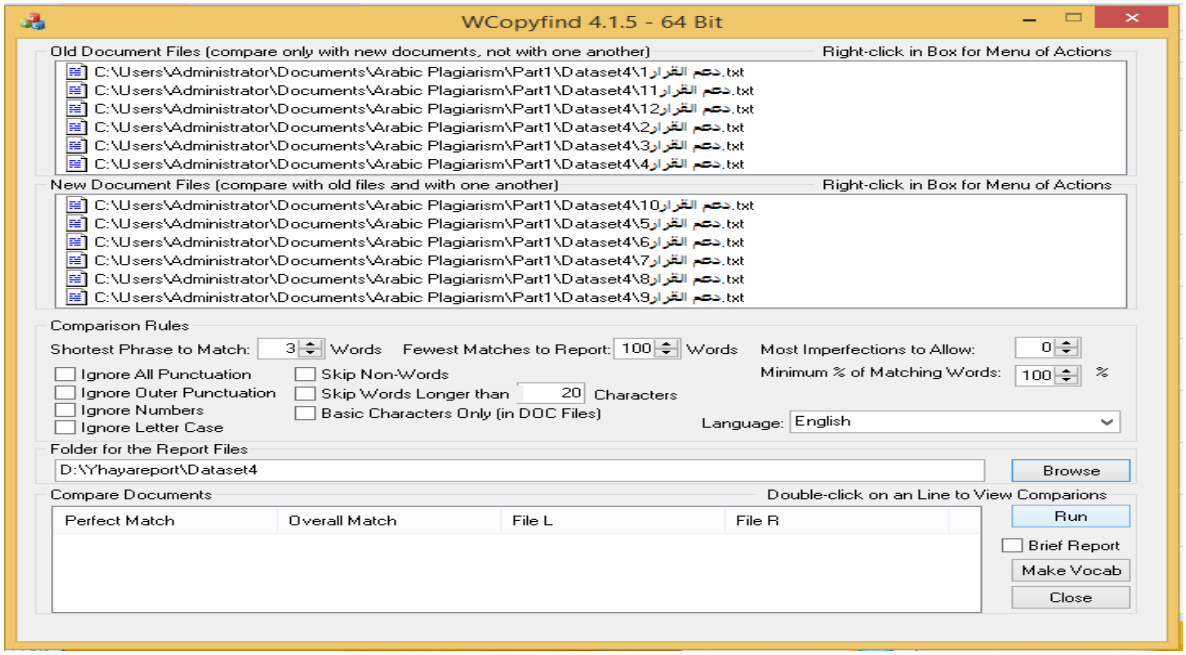
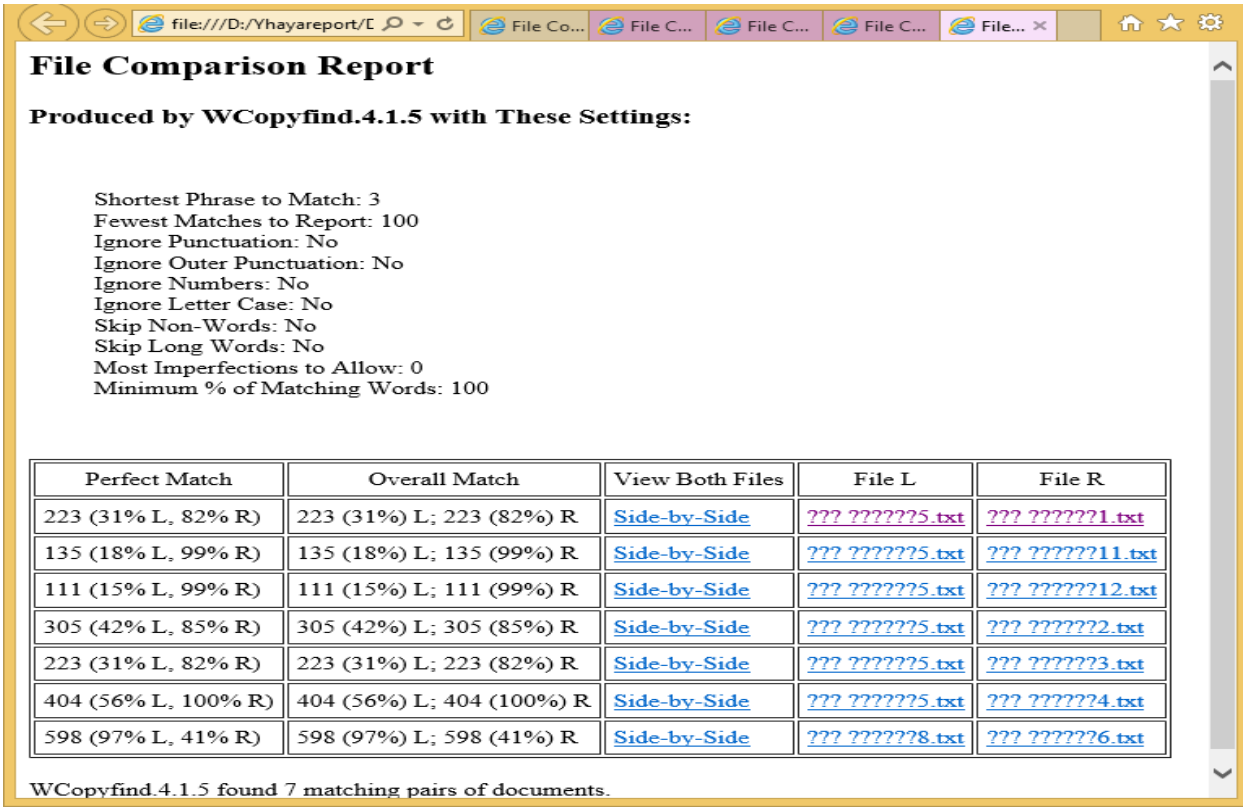


Figure 6.12 : Wcopyfind 4.4.1.5 Application uploaded Arabic files Dataset4 for checkup



File Comparison Report

Produced by WCopyfind.4.1.5 with These Settings:

Shortest Phrase to Match: 3
Fewest Matches to Report: 100
Ignore Punctuation: No
Ignore Outer Punctuation: No
Ignore Numbers: No
Ignore Letter Case: No
Skip Non-Words: No
Skip Long Words: No
Most Imperfections to Allow: 0
Minimum % of Matching Words: 100

Perfect Match	Overall Match	View Both Files	File L	File R
223 (31% L, 82% R)	223 (31%) L; 223 (82%) R	Side-by-Side	??? ???????5.txt	??? ??????1.txt
135 (18% L, 99% R)	135 (18%) L; 135 (99%) R	Side-by-Side	??? ???????5.txt	??? ??????11.txt
111 (15% L, 99% R)	111 (15%) L; 111 (99%) R	Side-by-Side	??? ???????5.txt	??? ??????12.txt
305 (42% L, 85% R)	305 (42%) L; 305 (85%) R	Side-by-Side	??? ???????5.txt	??? ???????2.txt
223 (31% L, 82% R)	223 (31%) L; 223 (82%) R	Side-by-Side	??? ???????5.txt	??? ???????3.txt
404 (56% L, 100% R)	404 (56%) L; 404 (100%) R	Side-by-Side	??? ???????5.txt	??? ???????4.txt
598 (97% L, 41% R)	598 (97%) L; 598 (41%) R	Side-by-Side	??? ???????8.txt	??? ???????6.txt

WCopyfind.4.1.5 found 7 matching pairs of documents.

Figure 6.13: Present Wcopyfind Application Report plagiarized on Dataset4

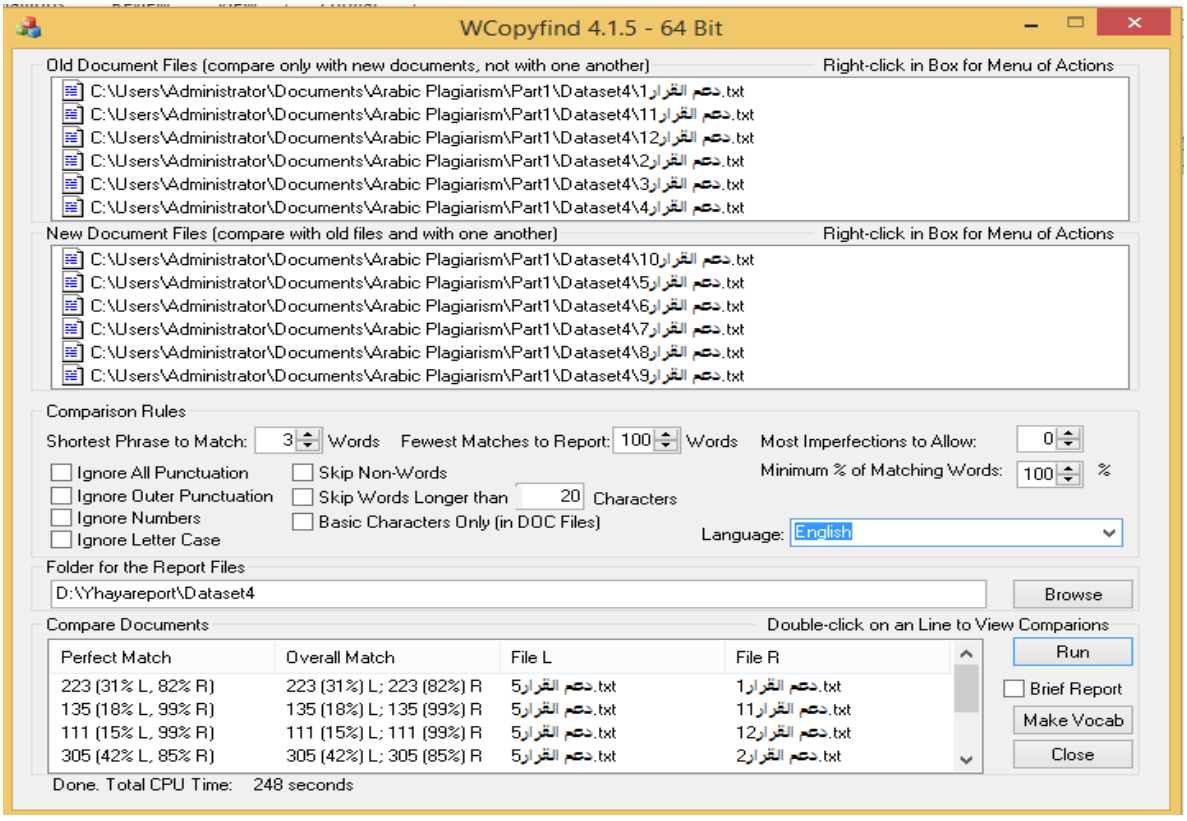


Figure 6.14 : Wcopyfind 4.4.1.5 Application plagiarized on Dataset4

6.6 Comparison between ADPDM Results andWCopyfind64.4.1.5 Tools

In this paragraph we would like to make a compare the tested performance between our tool "ADPDM" and WCopyfind 4.1.5 application on same datasets that mentioned above in term of their performance in detecting the plagiarism of the Arabic documents and the time taken. Table 6.10 and figure 6.15 as showing below.

Table 6.10: The comparison result between "ADPDM" and WCopyfind 4.4.1.5

	ADPDM		WcopyFind 4.15	
Datasets	Percentage Detection	Time in Second	Percentage Detection	Time in Second
Datasets1	14%	501	0%	135
Datasets2	8%	1374	0%	475
Datasets3	18%	1430.45	6.33%	271
Datasets4	94%	682.79	84%	357

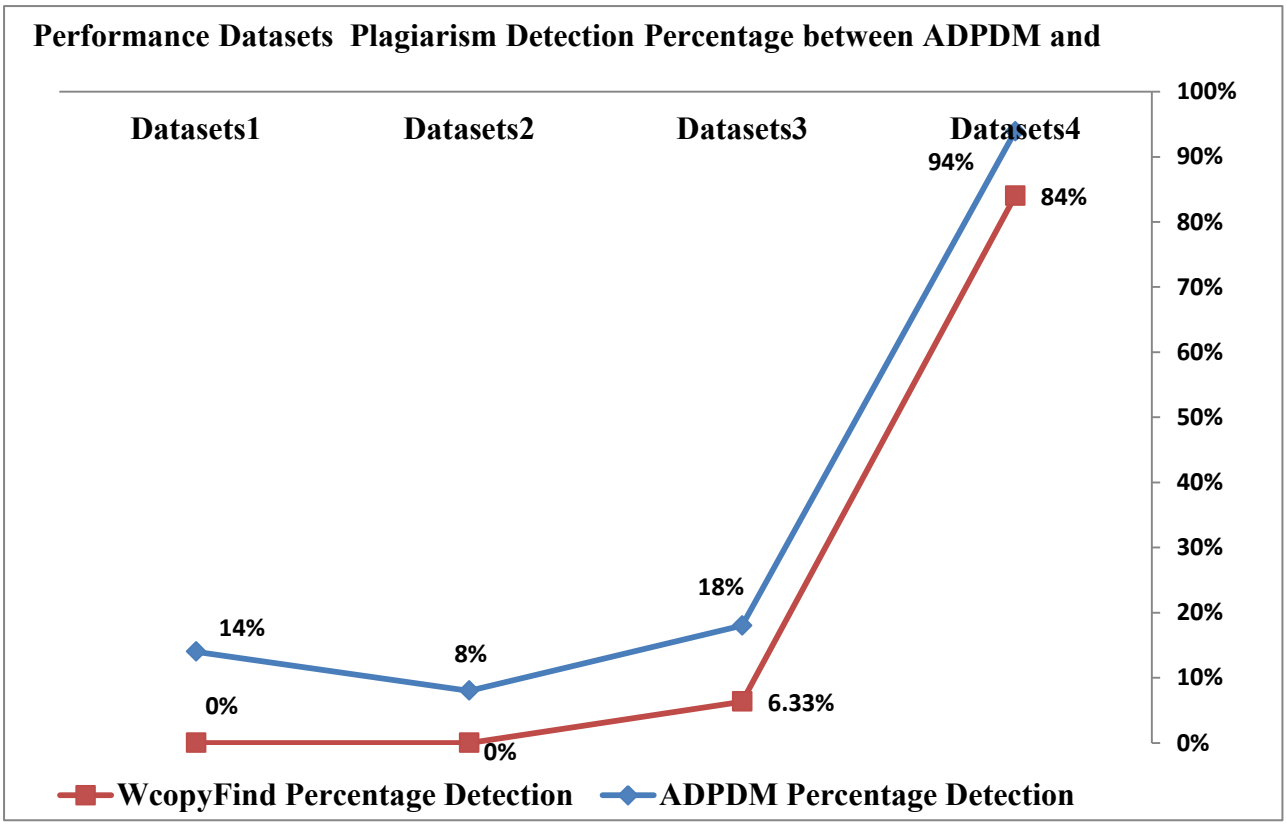


Figure 6.15: The performance of datasets plagiarism detection percentage between ADPDM and WcopyFind

The following figure 6.16 and table 6.10 discuss the compersion between our tool "ADPDM" and WCopyfind 4.1.5 application on same datasets that mentioned above in term of their performance in taken duoring plagiarism dectection of the Arabic documents. The result shows the time consuming to detection plagiraised by apply ADPDM toole on dataset1 are 501 second while WcopyFind 4.1.5 present 135 second , dataset2 take 1374 second in ADPDM , 475 second Wcompyfind 4.1.5 .in dataset3 the time present 1430.45 sencond in ADPDM tool, where Wconpyfind 4.1.5 get 271 second .Finaly ADPDM on dataset4 shows 681.79 second time taken while Wcopyfind 4.1.5 present 357 second.

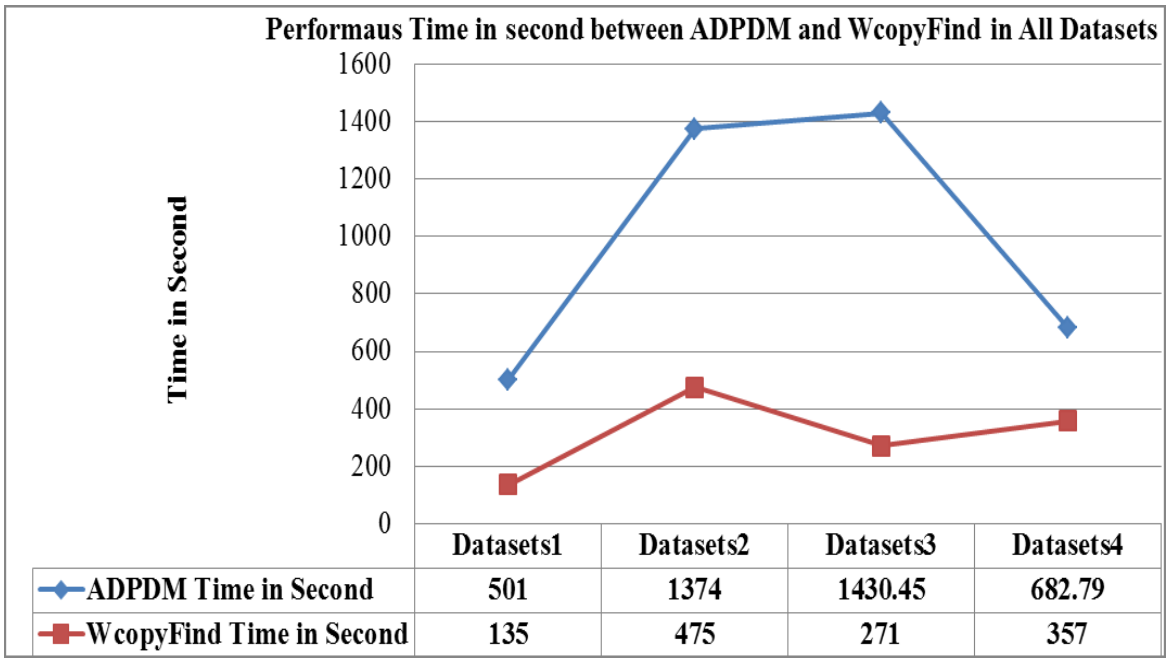


Figure 6.16 The compersion performance between ADPDM and WcopyFind time taken in second

6.7 Evaluation Measures

We evaluate through apply recall, precision, and f-measure important measures in the efficiency of the plagiarism detection as mention in chapter 5. As showing in table 6.11 present our datasets performans in the three measures and figure 6.17 as well.

Table 6.11 The datasets performans in the Recall, Precision and F-Measure

	ADPDM			
	Dataset1	**Daraset2**	**Dataset3**	**Dataset4**
Recall	0.566667	0.766666667	0.80	0.988071579
Precision	1	1	1	0.977056537
F-Measure	0.723404	0.867924528	0.8888889	0.982533187

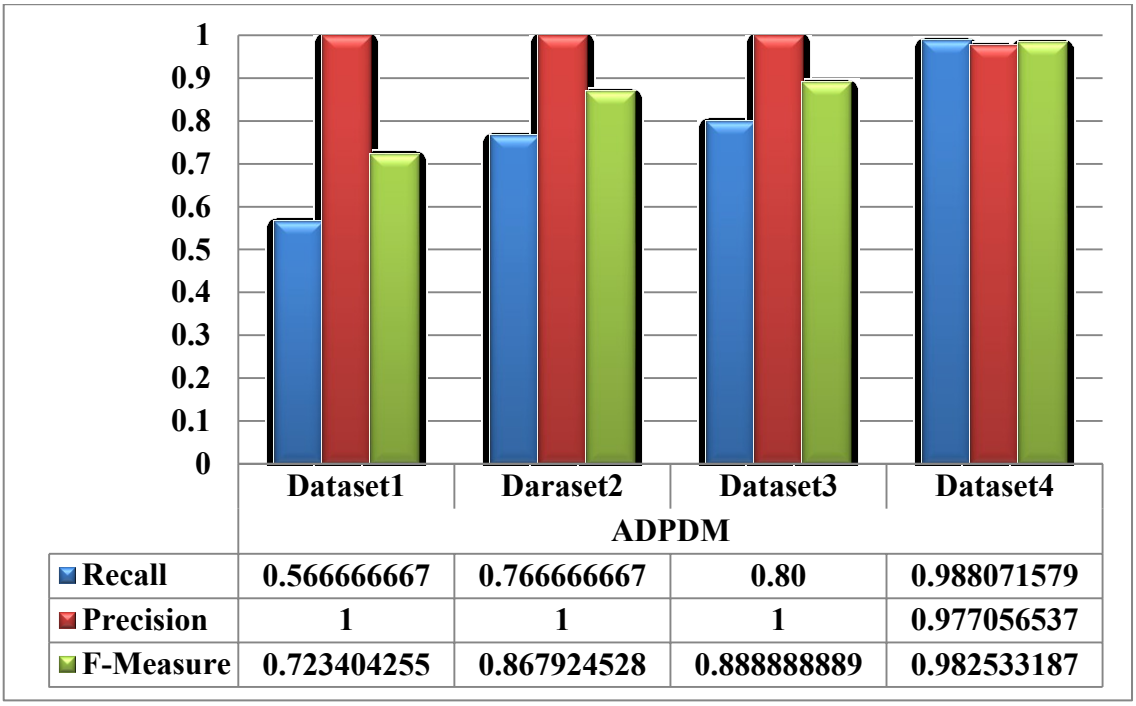


Figure 6.17 The dataset performans in the Recall, Precision and F-Measure

Also we apply a compersion evaluation between ADPDM and diffrent AraPlagDet tool through important measures in the efficiency of the plagiarism detection recall , recall, precision, and f-measure in dataset1 ,dataset2 and dataset3. As showing in table 6.12 and figure 6.18 are present the compersion evaluation between ADPDM and diffrent AraPlagDet tool[31] . Where recall and precision results were taken from a source AraPlagDet website [31]. Through the application of recall ,precision we conclude that evalutation result shows Magooda_2 is beast on recall 0.8314955 , Precision 0.8521183 and 0.84168059 F-Measure in the first rank. In the second rank

comes the ADPDM with recall 0.71 , Precision 1 , F-Measure 0.831168831. Palkovskii_1 comes in the third rank recall 0.5422843 , Precision 0.9774681 and F-Measure 0.697568373 more details as showing on table 6.12 and figure 6.18.

Table 6.12 The compersion evaluation between ADPDM and diffrent AraPlagDet tool[31]

	ADPDM	Basel ine	Palkovskii_1	Alzahrani	Magooda_2
Recall	0.71	0.5349007	0.5422843	0.530459	0.8314955
Precision	1	0.990391	0.9774681	0.830882	0.8521183
F-Measure	0.831168831	0.6946354	0.697568373	0.647521	0.84168059

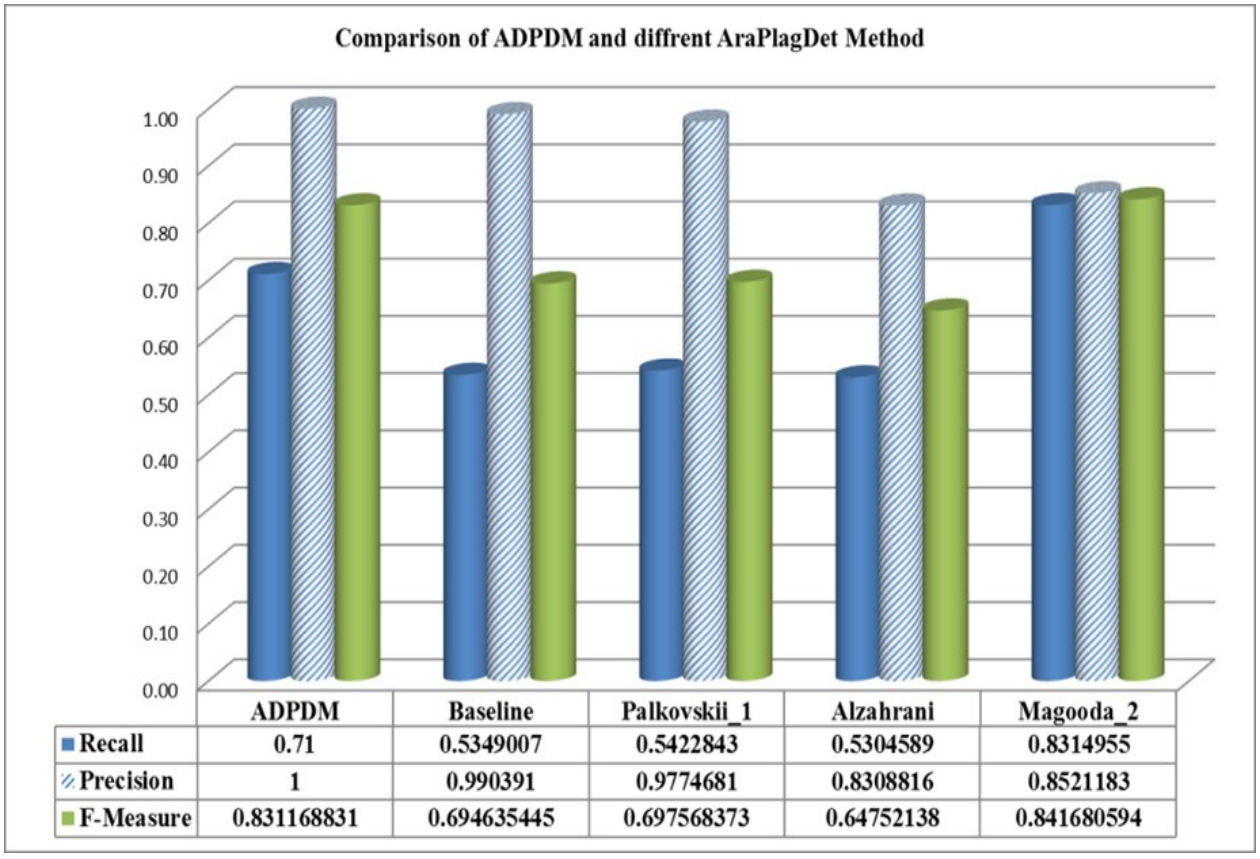


Figure 6.18: The compersion evaluation between ADPDM and diffrent AraPlagDet tool

6.8 Summary

In summary, this research aims at develop and implement the proposed “ADPDM” Arabic documents plagiarism detection model tools based on the introduced model that is capable in detecting plagiarism in Arabic documents. The preliminary experiments were carried out that our tools ADPDM tools it can detected Arabic document plagiarized. Then, we summarized Datasets Information Details (Corpus) which tested by ADPDM. This tool has gave honorable results compared to Wcopyfind in All datasets on the detect plagiarism in Arabic document. On the other hand the time spent to get the result we find Wcopyfind faster than ADPDM. Wcopyfind doesn’t support UTF file format.

CHAPTER VII

CONCLUSION AND FUTURE WORK

7.1 Introduction

This chapter presents a concluding remark for the work done to meet the project objectives. The main goal of this research is to develop and implement the proposed "ADPDM" Arabic documents plagiarism detection model tools based on the introduced model that is capable in detecting plagiarism in Arabic documents and search mechanism for the similar candidate documents within the corpus collection. The second goal of the logical document representation is to save computation time by avoiding unnecessary comparisons. For that reason we define a heuristic algorithm for each level in the tree: document level, paragraph level, and sentence level. We measure it using the Longest Common Substring (LCS) metric. In Chapter 4, we explained the framework that contain of six stages to detected Arabic document. In Chapter 5 we development of plagiarism detection tool and user interface for Arabic documents tested and analyzed results obtained from both. And here, findings and contributions of the study will illustrate the gap bridged by this research.

7.2 Findings

Our main finding is Arabic plagiarism best can be handled using heuristic Algorithm approach since it can cover more practices of plagiarism for Arabic language by using the Brian Kernighan and Dennis Ritchie (BKDR) hash function for chunk (3-gram) hashing. the logical document representation is to save computation time by avoiding unnecessary comparisons. For that reason we define a heuristic algorithm for each level in the tree: document level, paragraph level, and sentence level. We measure it using the Longest Common Substring (LCS) metric. In this study, preliminary experiments were carried out using our tool ADPDM and WCopyFind. The result shows that in dateset1 14% plagiarize detection during 501 second where WCopyFind detected 0% in 135 second, dataset2 shows 8% in 1374 second where WCopyFind detected 0% in 475 second , And also dataset3 18% plagiarize detection during 1430 second where WCopyFind detected 6.33% in 271 second , dataset4 94% plagiarize in1682.79 second where WCopyFind detected 81.44% in 357 second. The main conclusion is that ADPDM best resulthandled plagiarism detection while it is weak in

the time taken and WCopyFind it is weak to handled plagiarism detection while it best in the time taken.

7.3 Future Work

The future work might include, but not limited to, the following.

i. Enlarging the corpora and query to have thousands of documents.
ii. Upgrading the APDM to increase speed for detection plagiarism in Arabic documents by addition another method like Genetic Algorithm (GA).
iii. Enhance of detection Plagiarism report to include all spoofed files in a single detailed report
iv. Integrate the Web page with the Java program to be a single integrated module

7.4 Conclusions

Based on the literature review, introduced an Arabic documents plagiarism detection model, the framework for the introduced model that is capable in detecting plagiarism in Arabic documents and preliminary experiments performed in this study. The main conclusion is that ADPDM best result handled plagiarism detection in arabic document while it is weak in the time taken and WCopyFind it is weak to handled plagiarism detection while it best in the time taken. After apply the recall, precision, and f-measure to measures in the efficiency of the Arabic plagiarism detection in ADPDM the result shows recall average of all dataset is 0.780351, precision shows 0.994264 in average of all datasets and for f-measure is harmonic mean of precision and recall , result shows 0.865688 of f-measure. Then I made a comparison between the results that have obtained on ADPDM tool and the results that achevied by AraPlaDet 2015 tested different tool. The recall, precision and f-measure equations applied to those tools in same datasets. We conclude the comparative that our tool very impressive results cames in second rank with recall 0.71 , Precision 1 , F-Measure 0.831168831. Forthanmore, Magooda_2 is beast on recall 0.8314955 , Precision 0.8521183 and 0.84168059 F-Measure in the first rank.

REFERENCS

[1] Yahya. A. Abdelrahman , A. Khalid and I. M. Osman," A SURVEY OF PLAGIARISM ETECTION FOR ARABIC DOCUMENTS", INTERNATIONAL JOURNAL OF ADVANCED OMPUTER TECHNOLOGY VOLUME 4, NUMBER 6, (Page no:34-38) December 2015.

[2] Ameera Jadalla and Ashraf Elnagar "A Plagiarism Detection System for Arabic Text-Based Documents", Department of Computer Science, University of Sharjah, P.O. Box 27272, Sharjah, UAE, © Springer-Verlag Berlin Heidelberg 2012.

[3] Alaa m. Riad , farahat f. Farahat , aziza s. Asem & mahmoud a. Zaher,"Studying Different Methods For Plagiarism Detection", International Journal of Com-puter Science and Engineering (IJCSE) ISSN(P): 2278-9960; ISSN(E): 2278-9979 Vol. 2, Issue 5, Nov 2013, 147-154 © IASET.

[4] Cecillia Barnbaum, "Plagiarism: A Student's Guide to Recognizing It and Avoiding It.",ValdostaStateUniversity,https://mypages.valdosta.edu/cbarnbau/personal/teaching_MISC/plagiarism.htm (accessed October 10, 2017)

[5] Dumais S.T. Latent Semantic Analysis [J]. Annual Review of Information Science and Technology, 2005: 38-188, doi:10.1002/aris. 1440380105.

[6] Maurer H., Kappe F. , and Zaka B. ,Plagiarism- A survey. Journal of Universal Computer Science 12,8, 1050-1084, Aug. 2006.

[7] Fernando Sanchez-Vega, Esaú Villatoro-Tello, Manuel Montes-y, Luis Villase, Paolo Rosso, "Determining and characterizing the reused text for plagiarism detection" , Contents lists available at SciVerse ScienceDirect , Expert Systems with Applications 40 (2013) 1804–1813.

[8] S. Schleimer, D. Wilkerson, A. Aiken, "Winnowing: Local algorithms for document fingerprinting," In Proc. of the 2003 ACM SIGMOD Int. Conf. on Management of Data, San Diego, California, USA, 9-12 June 2003.

[9] G. Oberreuter, and J. D. Velsquez, "Text mining ap-plied to plagiarism detection: The use of words for detecting deviations in the writing style", Contents lists available at SciVerse ScienceDirect, Expert Sys-tems with Applications 40 (2013) 3756–3763.

[10] Imtiaz Hussain Khan, Muazzam Ahmed Siddiqui, Kamal Mansoor Jambi and Abobakr Ahmed Bagais, " A FRAMEWORK FOR PLAGIARISM DETEC-TION

IN ARABIC DOCUMENTS", Dhinaharan Nagamalai et al. (Eds) : CSEA, DKMP, AIFU, SEA – 2015 pp. 01–09, 2015. © CS & IT-CSCP 2015.

[11] Mohamed El Bachir Menai, Manar Bagais," APlag: A Plagiarism Checker for Arabic Texts" Depart-ment of Computer Science CCIS - King Saud Univer-sity Computer Science & Education (ICCSE 2011) August 3-5, 2011. SuperStar Virgo, Singapore

[12] J. A. Faidhi and S. K. Robison, "An empirical ap-proach for detecting program similarity within a uni-versity programming environment,"Computers and Education, 2008.

[13] K. Omar, B. Alkhatib, M. Dashash ,"The Implemen-tation of Plagiarism Detection System in Health Sci-ences Publications in Arabic and English Languages" International Review on Computers and Software (I.RE.CO.S.), Vol. 8, N. 4 ISSN 1828-6003 April 2013.

[14] S. M .Alzahrani, N. Salim, "Statement-based fuzzy-set IR versus fingerprints matching for plagiarism detection in Arabic documents," In Proc. of the 5th Postgraduate Annual Research Seminar (PARS09), Johor Bahru, Malaysia, 2009.

[15] L. Romans, G. Vita, and G. Janis, "Computer-based plagiarism detection methods and tools: an over-view", the 2007 international conference on Comput-er systems and technologies. 2007, ACM: Bulgaria.

[16] Bensalem. I., Boukhalfa, I., Rosso, P., Abouenour L., Darwish, K., A., & Chikhi, S. (2015). Overview of the AraPlagDet PAN@Fire2015 shared task on Arabic plagiarism detection. In fire2015 working notes papers. Gandhinger, India Vol. 114-117

[17] M. El Bachir Menai, " Detection of Plagiarism in Arabic Documents", I.J. Department of Computer Science, College of Computer and Information Sciences, King Saud University Saudi Arabia, 10, 80-89 Published Online September 2012 in MECS (http://www.mecs-press.org/)DOI:10.5815/ijitcs. 2012.10.10.

[18] Chow Kok Kent and Naomie. Salim, "Features based text similarity detection," arXiv preprint arXiv:1001.3487, 2010.

[19] M. Menai,and M. Bagais, "APlag: A Plagiarism Checker for Arabic Texts" The 6th International Conference on Computer Science & Education (ICCSE 2011), IEEE 2011. Depart-ment of Computer Science CCIS - King Saud Univer-sity Computer Science & Education (ICCSE 2011) August 3-5, 2011. SuperStar Virgo, Singapore

[20] Plagiarism.org. “What is Plagiarism?” Web 4 Nov. 2015. <http://www.plagiarism.org/ plagiarism-101/what-is-plagiarism>.

[21] Randa. K., “A Plagiarism Detection Tool For Arabic Text Document”, Thesis of Master Degree ,Sudan University of Science and Technology , 2010, Sudan.

[22] Shivakumar N., Garcia-Molina H. SCAM: a copy detection mechanism for digital documents [C]. In: Proceedings of the 2nd International Conference on Theory and Practice of Digital Libraries, Aus-tin,Texas, USA, June 1995.

[23] Salha Mohammed Alzahrani, and Naomie Salim, "Plagiarism Detection In Arabic Scripts Using Fuzzy Information Retrieval", Proceedings of 2008 Student Conference on Research and Development (SCOReD 2008), 26-27 Nov. 2008, Johor, Malaysia.

[24] S. M. Alzahrani, N. Salim, “Statement-based fuzzy-set IR versus fingerprints matching for plagiarism de-tection in Arabic documents,” In Proc. of the 5th Postgraduate Annual Research Seminar (PARS09), Johor Bahru, Malaysia, 2009.

[25] Types of Plagiarism (n.d.) Retrieved Oct. 2, 2009, from http://www.plagiarism.org/plag_article_ types_ of _plagiarism.html Reprinted with permission.

[26] http://www.plagiarism.com/, visited 30 Apr 2018.

[27] UKessays.com, "A Survey Of Plagiarism Detection Methods Information Technology Essay.". 11 2013.

[28] http://www.turnitin.com/ visited ,December 12, 2016

[29] Seo, J., Croft, W.B.: Local text reuse detection. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieva, pp. 571–578 (2008)

[30] Alzahrani SM, Salim N. Statement-based fuzzy-set IR versus fingerprints matching for plagiarism detection in Arabic documents. In Proc. of the 5th Postgraduate Annual Research Seminar, Malaysia; 2009.

[31] Gruner G., Naven S. Tool support for plagiarism detection in text documents [C]. In: Proceedings of the ACM symposium on Applied Computing, Santa Fe, New Mexico, 2005, 13-17.

[32] Ramesh R. Naik , Maheshkumar B. Landge and C. Namrata Mahender “A Review on Plagiarism Detection Tools”, International Journal of Computer Applications (0975 – 8887) Volume 125 – No.11, September 2015

[33] Types of Plagiarism: You Can't Avoid it if You Don't Know What it is<https://unicheck.com/blog/types-of-plagiarism> University Survival Kit 2015/08/17. Copyright © 2018 P1K, LTD. All rights reserved

[34] WEBER WULFF, D. Copy, Shake, and Paste - A blog about plagiarism from a German professor, written In English. Online Source. Retrieved Nov. 28, 2010 from: http://copyshake-paste.blogspot.com, , Nov. 2010

[35] Stefan, G., & Stuart, N. (2005). Tool support for plagiarism detection in text documents. Paper presented at the Proceedings of the 2005 ACM symposium on Applied computing

[36] B. Belkhouche, A. Nix, and J. Hassell, "Plagiarism detection in software designs," Proc. of the 42nd An-nual Southeast Regional Conference, 2004.

[37] Liles, Jeffrey A. and Michael E. Rozalski., "It's a Matter of Style: A Style Manual Workshops for Preventin Plagiarism.", College & Undergraduate Libraries, 11 (2) , p. 91-101, 2004.

[38] Farahat F. Farahat1, Aziza S. Asem2, Mahmoud A. Zaher3*and Ahmed M. Fahiem4, "Detecting Plagia-rism in Arabic E-Learning Using Text Mining"British Journal of Mathematics & Computer Science 8(4): 298-308, 2015, Article no.BJMCS.2015.163 ISSN: 2231-0851.

[39] David Dowty. Thematic Proto-Roles and Argument Selection. Language, Vol. 67, No. 3. (Sep., 1991), pp. 547-619.

[40] Bretag T., and Mahmud S. , self-plagiarism or Appropriate Textual Re-use, Journal of Academics Ethics 7, 193-205, 2009

[41] Y. A. Abdelrahman, et al., "A Method For Arabic Documents Plagiarism Detection," International Journal of Computer Science and Information Security, vol. 15, p. 79, 2017.

[42] C. Hoad, J. Zobel, "Methods for identifying versioned and plagiarized documents," Journal of the American Society for Information Science and Technology, vol. 54, no. 3, pp. 203-215, 2003.

[43] G. Whale, "Plague : plagiarism detection using pro-gram structure,"Dept. of Computer Science Technical Report 8805, University of NSW,Kensington, Aus-tralia, 2008

[44] U. Manber, "Finding similar files in a large file system," Winter USENIX Technical Conf., San Francisco, CA, USA, 1994.

[45] Izzat Alsmadi1 , Ikdam AlHami2 and Saif Ka-zakzeh3," Issues Related to the Detection of Source Code Plagiarism in Students Assignments", Interna-tional Journal of Software Engineering and Its Appli-cations Vol.8, No.4 (2014), pp.23-34.

[46] L. Prechelt, G. Malpohl, and M. Philippsen, “Finding plagiarisms among a set of programs with JPlag,” Journal of Universal Computer Science, 2008.

[47] Asim M. El Tahir Ali, Hussam M. Dahwa Abdulla, and V´aclav Sn´aˇsel , ”Overview and Comparison of Plagiarism Detection Tools” Dateso 2011, pp. 161–172, ISBN 978-80-248-2391-1.

[48] S. M. Alzahrani and N. Salim, "On the use of fuzzy information retrieval for gauging similarity of arabic documents," in Applications of Digital Information and Web Technologies, 2009. ICADIWT'09. Second International Conference on the, 2009, pp. 539-544.

[49] M. Mozgovoy, K. Fredriksson, and D. White, “Fast plagiarism Detection system,” Lecture Notes in Com-puter Science, 2005.

[50] S. M. Alzahrani, et al., "Work in progress: Developing Arabic plagiarism detection tool for elearning systems," in Computer Science and Information Technology-Spring Conference, 2009. IACSITSC'09. International Association of, 2009, pp. 105-109.

[51] I. Bensalem, et al., "Intrinsic plagiarism detection in Arabic text: Preliminary experiments," in II Spanish Conference on Information Retrieval (CERI’12), 2012.

[52] Ramya l and Venkatalakshmi R., "Intelligent plagiarism detection," International Journal of Research in Engineering & Advanced Technology (IJREAT), vol. 1, pp. 171-174, 2013

[53] S. Ouamour and H. Sayoud, "Authorship attribution of short historical arabic texts based on lexical features," in Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2013 International Conference on, 2013, pp. 144-147

[54] A. S. Altheneyan and M. E. B. Menai, "Naïve Bayes classifiers for authorship attribution of Arabic texts," Journal of King Saud UniversityComputer and Information Sciences, vol. 26, pp. 473-484, 2014.

[55] A. F. Otoom, et al., "Towards author identification of Arabic text articles," in Information and Communication Systems (ICICS), 2014 5th International Conference on, 2014, pp. 1-4.

[56] http://plagiarism.bloomfieldmedia.com/wordpress/software/wcopyfind/visited ,December 12, 2016

[57] Si A., Leong H., Lau R. CHECK: a document plagiarism detection system [C]. In: Proceedings of ACM Symposium for Applied Computing, Feb. 1997, 70-77.

[58] http://www.canexus.com/eve/visited ,December 12, 2016

[59] ("Plagiarism", 2007).< http://www.plagiarism.org/article/what-is-plagiarism> Published May 18, 2017. (Accessed: November 13, 2017)

[60] Noble Nnamani, AphriaPUB Ltd Ways to easily avoid Plagiarism in Research Papers Published on November 13, 2017 https://www.linkedin.com/pulse/6-ways-easily-avoid-plagiarism-research-papers-aphriapub-ltd (Accessed: November 13, 2017)

[61] Encyclopedia Britannica Online.) <https://hwarmstrong.com/allen-armstrong/index.htm> (Encyclopædia Britannica. 2017. Encyclopædia Britannica Online. 13 Mar. 2017

[62] Deva, J. J. G., Carroll, N. L., & Calvo, R. A. (2006). Applying Plagiarism Detection to Engineering Education. Paper presented at the 7th International Conference on Information Technology Based Higher Education andTraining (ITHET '06).

[63] Lyon, C., Barrett, R., & Malcolm, J. (2006). Plagiarism is Easy, but also Easy To Detect. Plagiary: Cross-Disciplinary Studies in Plagiarism, Fabrication, and Falsification., I, 57-65.

[64] Boubaker Kahloula , Jawad Berri ,.(2016)"Plagiarism Detection in Arabic Documents: Approaches, Architecture and Systems", Journal of Digital Information Management ‰ Volume 14 Number 2 ‰ April 2016

[65] <http://en.writecheck.com/ways-to-avoid-plagiarism/> Ways to Avoid Plagiarism in Research Copyright © 2017 Turnitin, LLC. All rights reserve,17/12/2017 accsessd .

[66] The Writing Center @the university of Wisconsin-madison https://writing.wisc.edu/Handbook/QPA_plagiarism.html Source

https://writing.wisc.edu/Handbook/Acknowledging_Sources.pdf accssed October 2017.

[67] Types of Plagiarism You Should Be Aware of." Types of Plagiarism: What Are They and How to Avoid Them. Www.unplag.com, n.d. Web. 22 Aug. 2016

[68] "Ways to Avoid Plagiarism." Plagiarism Checker. Www.turnitin.com, n.d. Web. 22 Aug. 2016.

[69] Alzahrani, Salha, Salim, Naomie., Kent, Chow Kok., Binwahlan,Mohammed Salem., Suanmali, Ladda. (2010).The development of cross-language plagiarism detection tool utilising fuzzy swarm-based summarization, In: Intelligent Systems Design and Applications (ISDA), 10th International Conference on, p. 86–90. IEEE.

[70] Hussein, Abdullah Al, Jayabrabu, R., Tirumalai, Saravanan Venkataraman (2010).Detection of plagiarism in arabic texts using text mining: A software agent based approach, In: Intelligent Systems Design and Applications (ISDA), 10th Computational Linguistics 1(1).

[71] RapidMiner Inc. (2016). Rapidminer, Online, Cited: January 13, https://rapidminer.com.

[72] Darwish, Kareem., Walid Magdy. (2014). Arabic information retrieval. Foundations and Trends, In: Information Retrieval, 7. 239–342, April.

[73] Boukhatem, Nadera,.The Arabic natural language processing: Introduction and challenges, International Journal of English Language & Translation Studies, 2. El Tarf University 2(3), 106-112 Retrieved from http://www.eltsjournal.org

[74] A. Jadalla and A. Elnagar, "A fingerprinting-based plagiarism detection system for Arabic text-based documents," in Computing Technology and Information Management (ICCM), 2012 8th International Conference on, 2012, pp. 477-482.

[75] HEINTZE, Nevin. Scalable document fingerprinting.Proceedings of the Second USENIX Workshop on Electronic Commerce. Oakland, California. 1996.

[76] Khoja'S.Stemming Arabic Text [R]. 2001 Pacific University. Visited on 13 May 2015 http://zeus.cs.pacificu.edu/shereen/research.htm.

[77] Black W., Elkateb S., Rodriguez H., Alkhalifa M., Vossen P., Pease A., Fellbaum C. Introducing the Arabic WordNet project [C]. In: Proceedings of the 3rd International WordNet Conference, Masaryk University, Brno, 2006, 295-300.

[78] Intisar Abakush ,"METHODS AND TOOLS FOR PLAGIARISM DETECTION IN ARABIC DOCUMENTS" Intisar Abakush Singidunum University, 32 Danijelova Street, Belgrade, Serbia INTERNATIONAL SCIENTIFIC CONFERENCE ON ICT AND EBUSINESS RELATED RESEARCH SINTEZA 2016.

[79] Bela Gipp. Citation-based plagiarism detection. In Citation-based Plagiarism Detection, pages 57– 88, Springer, 2014.

[80] Salha M Alzahrani, Naomie Salim, and Ajith Abraham. Understanding plagiarism linguistic patterns, textual features, and detection methods. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 42(2):133–149, 2012.

[81] AS Bin-Habtoor and MA Zaher. A survey on plagiarism detection systems. International Journal of Computer Theory and Engineering, 4(2):185, 2012.

[82] Asushi Ogawa, Tetsuya Morita, and Kiyohiko Kobayashi. A fuzzy document retrieval system using the keyword connection matrix and a learning method. Fuzzy sets and systems,39(2):163–179, 1991.

[83] Hermann A Maurer, Frank Kappe, and Bilal Zaka. Plagiarism a survey. J. UCS, 12(8):1050–1084,2006.

[84] Salha Alzahrani and Naomie Salim Fuzzy semantic-based string similarity for extrinsic plagiarism detection: Lab report for PAN at CLEF'10," presented at the 4th Int. Workshop PAN-10, Padua, Italy, 2010.

[85] Ashraf S Hussein. A plagiarism detection system for arabic documents. In Intelligent Systems' 2014, Springer International Publishing, 2015. p. 541-552.

[86] Ahmed Magooda, Ashraf Y Mahgoub, Mohsen Rashwan, Magda B Fayek, and Hazem M Raafat.Rdi system for extrinsic plagiarism detection (RDI-RED), working notes for panaraplagdet at fire 2015. In FIRE Workshops, pages 126–128, 2015.

[87] Salha Alzahrani. Arabic plagiarism detection using word correlation in n-grams with koverlapping approach, working notes for panaraplagdet at fire 2015 workshops, p 123–125, 2015.

[88] McCandless, M., Hatcher, E., & Gospodnetic, O. (2010). Lucene in Action: Covers Apache Lucene 3.0. Manning Publications Co. 89 Salton, G. (1989). Automatic text processing: The transformation, analysis, and retrieval of. Reading: Addison-Wesley.

[89] Sabrina Simmons and Zachary Estes. Using latent semantic analysis to estimate similarity. In Proceedings of the Cognitive Science Society, pages 2169–2173, 2006

[90] Zdenek Ceska. Plagiarism detection based on singular value decomposition. In Advances in natural language processing, pages 108–119. Springer, 2008.

[91] http://www.canadianarabcommunity.com/arabiclanguage.php

APPENDIX A

Khoja's Arabic Stop words List, Pacific University

أمامهن	إنكما	الا	أضحى	هنا	لم	حين	ان
أيضا	باللذان	فكان	اضحى	وقد	هؤلاء	ومن	بعد
بأيهم	بأي	ستكون	ظل	كانت	فإن	لا	ضد
اخرى	اجل	مما	مابرح	لذلك	فيه	ليسب	يلي
اللذان	اللاتي	أبو	مافتئ	أمام	ذلك	وكانت	الى
آخر	اياه	بإن	ماانفك	هناك	لو	أي	في
أمامي	امامها	الذي	بات	قبل	عند	ما	من
أيضا	أيا	اليه	صار	معه	اللذين	عنه	حتى
بامكان	إنما	يمكن	ليس	يوم	كل	حول	وهو
بأيهما	باللذين	بهذا	إن	منها	بد	دون	يكون
اخيرا	بأيا	لدي	كأن	إلى	لدى	مع	به
أبدا	احدا	وأن	ليت	إذا	وثي	لكنه	وليس
أين	اللاحق	وهي	لعل	هل	أن	ولكن	أحد
إنها	ايضا	وأبو	لاسيما	حيث	ومع	له	على
إنهم	امامهم	آل	ولايزال	هي	فقد	هذا	وكان
بان	أيان	الذي	الحالي	اذا	بل	والتي	تلك
بأيهن	إننا	هن	ضمن	او	هو	فقط	كذلك
اذ	باللواتي	الذى	اول	و	عنها	ثم	التي
أثناء	بأية	انكما	وله	ما	منه	هذه	وبين
أنا	احدى	الغير	ذات	لا	بها	أنه	فيها
أينما	اللتان	اولئك	اي	الي	وفي	تكون	عليها
إنهما	اين	أمامنا	بدلا	إلي	فهو	قد	إن
بانه	امامهما	أى	اليها	مازال	تحت	بين	وعلى
بإحدى	أية	باللتين	انه	لازال	لها	جدا	لكن
اذا	إنني	بأى	الذين	لايزال	أو	لن	عن
اللواتي	بالنسبة	اثناء	فانه	مايزال	إذ	نحو	مساء

اجل	بأيها	القادم	وان	اصبح	علي	كان	ليس
أنت	احيانا	اي	والذي	أصبح	عليه	لهم	منذ
أيها	اللتين	أمامه	وهذا	أمسى	كما	لأن	الذي
إنهن	ايها	أي	لهذا	امسى	كيف	اليوم	أما

APPENDIX B

Arabic Root Dictionary

أثر	أتي	أتو	أتن	أتم	أبي	أبو	أبه	أبن	أبل	أبق	أبط	أبض	أبر	أبد	أبب
أخو	أخذ	أخت	أحن	أحد	أحح	أجن	أجم	أجل	أجص	أجر	أجج	أجب	أثم	أثل	أثف
أرم	أرق	أرف	أرض	أرس	أرخ	أرج	أرث	أرب	أذي	أذن	أدي	أدو	أدم	أدد	أدب
أسل	أسس	أسر	أسد	أست	أزي	أزم	أزل	أزق	أزف	أزز	أزر	أزح	أزج	أزب	أرو
أفل	أفق	أفف	أطم	أطط	أطر	أصل	أصص	أصر	أصد	أشف	أشر	أشب	أسي	أسو	أسن
ألي	ألو	أله	ألم	ألق	ألف	ألس	ألب	أكم	أكل	أكف	أكر	أكد	أقي	أقق	أقت	أفن
أوب	أهل	أهب	أني	أنن	أنق	أنف	أنس	أنث	أنب	أمو	أمن	أمم	أمل	أمع	أمس	أمر
أيض	أيس	أير	أيد	أيب	أوي	أوه	أون	أول	أوق	أوف	أوض	أوش	أوز	أور	أود	أوج
													أين	أيم	أيل	أيك
بجس	بجر	بجد	بجح	بثق	بثر	بثث	بتل	بتك	بتع	بتر	بتت	ببر	ببج	بأس	بأز	بأر

بخو	بخل	بخق	بخع	بخس	بخر	بخخ	بخت	بحش	بحر	بحح	بحث	بحت	بجن	بجم	بجل	بجع
برج	برب	برأ	بزل	بذر	بذذ	بذخ	بذأ	بدي	بدو	بده	بدن	بدل	بدع	بدر	بدد	بدأ
بزق	بزغ	بزز	بزر	بري	بره	برم	برك	برق	برع	برض	برص	برش	برز	برر	برد	برح
بصق	بصص	بصر	بشم	بشك	بشع	بشش	بشر	بسم	بسل	بسق	بسط	بسس	بسر	بسأ	بزي	بزل
بظظ	بظر	بطي	بطن	بطم	بطل	بطق	بطط	بطش	بطر	بطخ	بطح	بطأ	بضع	بضض	بصم	بصل
بقق	بقع	بقر	بقج	بغي	بغل	بغض	بغش	بغر	بغت	بعل	بعق	بعض	بعر	بعد	بعج	بعث
بلف	بلغ	بلع	بلط	بلص	بلر	بلد	بلح	بلج	بكي	بكم	بكل	بكر	بكت	بكأ	بقي	بقل
بهل	بهق	بهظ	بهز	بهر	بهج	بهت	بني	بنن	بند	بنج	بلي	بلو	بله	بلم	بلل	بلق
بيت	بول	بوق	بوغ	بوص	بوش	بوس	بوز	بور	بوخ	بوح	بوت	بوب	بوأ	بهي	بهو	بهم
													بين	بيع	بيض	بيد
ترك	ترف	ترع	ترس	ترخ	ترح	ترب	تخم	تخخ	تحف	تجر	تبن	تبل	تبع	تبر	تبب	تأر
تلف	تلع	تلد	تكي	تكك	تقي	تقو	تقن	تفه	تفل	تفك	تفف	تفح	تعس	تعب	تسع	تره
تيم	تيل	تيس	تير	تيح	توه	توق	توج	توب	تهم	تنأ	تمم	تمر	تلو	تله	تلم	تلل
																تيه
ثري	ثرو	ثرم	ثرر	ثرد	ثرب	ثدي	ثخن	ثجج	ثبن	ثبق	ثبط	ثبر	ثبج	ثبت	ثأر	ثأب
ثلل	ثلج	ثلث	ثلب	ثكن	ثكل	ثقل	ثقف	ثقب	ثفن	ثفل	ثفر	ثغو	ثغم	ثغر	ثعل	ثعب
					ثوي	ثول	ثور	ثوب	ثني	ثنو	ثنن	ثمن	ثمم	ثمل	ثمر	ثلم

جحر	جحد	جثو	جثم	جثل	جثث	جبي	جبه	جبن	جبل	جبس	جبر	جبذ	جبح	جبب	جأش	جأر
جذب	جدي	جدو	جدل	جدف	جدع	جدر	جدد	جدح	جدث	جدب	جخف	جخخ	جحم	جحف	جحظ	جحش
جرض	جرش	جرس	جرز	جرر	جرذ	جرد	جرح	جرب	جرأ	جذو	جذم	جذل	جذف	جذع	جذر	جذذ
جسر	جسد	جسأ	جزي	جزم	جزل	جزف	جزع	جزز	جزر	جزد	جزأ	جري	جرن	جرم	جرف	جرع
جفف	جفر	جفت	جفأ	جعل	جعر	جعد	جعب	جصص	جشم	جشع	جشش	جشر	جشأ	جسو	جسم	جسس
جلي	جلو	جلم	جلل	جلق	جلف	جلط	جلص	جلس	جلد	جلخ	جلح	جلب	جكر	جفو	جفن	جفل
جنن	جنف	جنس	جنز	جند	جنح	جنب	جمم	جمل	جمك	جمع	جمش	جمس	جمر	جمد	جمخ	جمح
جوع	جوط	جوس	جوز	جور	جود	جوح	جوب	جهي	جهم	جهل	جهض	جهش	جهز	جهر	جهد	جني
					جيل	جيف	جيش	جير	جيب	جيأ	جوي	جوو	جون	جول	جوق	جوف
حتم	حتك	حتف	حتر	حتد	حتت	حبو	حبن	حبل	حبك	حبق	حبط	حبش	حبس	حبر	حبذ	حبب
حدج	حدث	حدب	حدأ	حجو	حجن	حجم	حجل	حجف	حجز	حجر	حجج	حجب	حثو	حثل	حثر	حثث
حرج	حرث	حرب	حذي	حذو	حذق	حذف	حذر	حدي	حدو	حدم	حدل	حدق	حدف	حدس	حدر	حدد
حزز	حزر	حزب	حري	حرو	حرن	حرم	حرك	حرق	حرف	حرض	حرص	حرش	حرس	حرز	حرر	حرد
حشم	حشك	حشف	حشش	حشر	حشد	حسو	حسن	حسم	حسك	حسس	حسر	حسد	حسب	حزن	حزم	حزق
حطم	حطط	حطب	حضن	حضض	حضر	حصي	حصو	حصن	حصل	حصف	حصص	حصر	حصد	حصب	حشي	حشو
حقق	حقر	حقد	حقب	حفي	حفو	حفن	حفل	حفف	حفظ	حفز	حفر	حفد	حظي	حظو	حظظ	حظر
حلي	حلو	حلم	حلل	حلك	حلق	حلف	حلس	حلج	حلب	حكي	حكم	حكك	حكر	حقو	حقن	حقل
حنث	حنت	حنأ	حمي	حمو	حمم	حمل	حمق	حمط	حمض	حمص	حمش	حمس	حمز	حمر	حمد	حمأ

حوس	حوز	حور	حوذ	حود	حوج	حوث	حوت	حوب	حوم	حوي	حيث	حيد	حير	حيز	حيص
حيض	حيط														
خثر	ختن	ختم	ختل	ختر	خبي	خبو	خبل	خبع	خبط	خبص	خبز	خبث	خبر	خبب	خبأ
خرج	خرت	خرب	خرأ	خذو	خذل	خذف	خذأ	خدن	خدم	خدل	خدع	خدش	خدر	خدد	خدج
خزم	خزل	خزق	خزف	خزع	خزز	خزر	خرم	خرق	خرف	خرع	خرط	خرص	خرس	خرز	خرر
خصر	خصب	خشي	خشن	خشم	خشف	خشع	خشش	خشر	خشت	خشب	خسف	خسس	خسر	خسأ	خزي
خطف	خطط	خطر	خطب	خطأ	خضم	خضل	خضع	خضض	خضر	خضد	خضب	خصي	خصم	خصل	خصف
خلس	خلد	خلج	خلب	خقن	خفي	خفق	خفف	خفض	خفش	خفس	خفر	خفت	خطي	خطو	خطم
خمل	خمع	خمص	خمش	خمس	خمر	خمد	خمج	خلي	خلو	خلل	خلك	خلق	خلف	خلع	خلط
خوض	خوص	خور	خوذ	خوخ	خوج	خني	خنو	خنن	خنق	خنف	خنع	خنص	خنس	خنث	خمن
					خيم	خيل	خيط	خيش	خيس	خير	خيب	خوي	خوو	خون	خول
دحر	دجو	دجن	دجل	دجر	دجج	دثر	دبل	دبك	دبق	دبغ	دبش	دبس	دبر	دبج	دبب
درع	درس	درز	درر	درد	درج	درب	درأ	دخي	دخن	دخل	دخس	دحو	دحل	دحض	دحش
دشن	دشش	دشر	دشت	دسو	دسم	دسس	دسر	دست	دري	درو	دره	درن	درم	درك	درق
دفر	دفأ	دغم	دغل	دغص	دغش	دغر	دعي	دعو	دعم	دعك	دعع	دعس	دعر	دعج	دعب
دلس	دلح	دلج	دلب	دكن	دكك	دقل	دقق	دقع	دقر	دفي	دفن	دفل	دفق	دفف	دفع

دمم	دمل	دمك	دمغ	دمع	دمس	دمر	دمج	دمث	دلي	دلو	دله	دلل	دلك	دلق	دلف	دلع
دهم	دهك	دهق	دهش	دهس	دهر	دني	دنو	دنن	دنق	دنف	دنس	دنر	دنأ	دمي	دمو	دمن
دوم	دول	دوك	دوق	دوف	دوغ	دوط	دوش	دوس	دور	دود	دوخ	دوح	دوب	دوأ	دهي	دهن
									دين	ديم	ديك	ديس	دير	ديث	دوي	دون
ذعف	ذعر	ذري	ذرو	ذرق	ذرف	ذرع	ذرر	ذرح	ذرب	ذرأ	ذخر	ذحل	ذبل	ذبح	ذبب	ذأب
ذهن	ذهل	ذهب	ذنب	ذمي	ذمم	ذمر	ذلل	ذلق	ذلف	ذكي	ذكو	ذكر	ذقن	ذفر	ذعن	ذعق
									ذيل			ذيع	ذوي	ذوق	ذود	ذوب
ربق	ربغ	ربع	ربط	ربض	ربص	ربد	ربح	ربت	ربب	ربأ	رأي	رأم	رأف	رأس	رأد	رأب
رثو	رثث	رتو	رتن	رتم	رتل	رتك	رتق	رتع	رتج	رتت	رتب	ربي	ربو	ربن	ربل	ربك
رحق	رحض	رحب	رجي	رجو	رجن	رجم	رجل	رجف	رجع	رجس	رجز	رجح	رجج	رجب	رجأ	رثي
ردم	ردف	ردغ	ردع	ردس	ردد	ردح	ردأ	رخي	رخو	رخم	رخص	رخخ	رحي	رحو	رحم	رحل
رسخ	رسح	رسب	رزي	رزن	رزم	رزق	رزغ	رزز	رزح	رزب	رزأ	رذل	رذذ	ردي	رده	ردن
رصع	رصص	رصد	رشو	رشن	رشم	رشق	رشف	رشش	رشد	رشح	رسو	رسن	رسم	رسل	رسف	رسغ
رعص	رعش	رعد	رعب	رطن	رطم	رطل	رطب	رضي	رضو	رضم	رضع	رضض	رضخ	رضب	رصن	رصف
رفح	رفث	رفت	رفأ	رغو	رغم	رغف	رغد	رغث	رغب	رعي	رعو	رعن	رعم	رعل	رعف	رعع
رقط	رقص	رقش	رقد	رقب	رقأ	رفو	رفه	رفل	رفق	رفف	رفع	رفض	رفص	رفش	رفس	رفد
رمث	ركو	ركن	ركم	ركل	ركك	ركع	ركض	ركس	ركز	ركد	ركب	رقي	رقن	رقم	رقق	رقع

رنق	رنخ	رنح	رمي	رمن	رمم	رمل	رمك	رمق	رمض	رمص	رمش	رمس	رمز	رمد	رمح	رمج
روج	روث	روب	رهو	رهن	رهم	رهل	رهق	رهف	رهط	رهص	رهج	رهب	رني	رنو	رنن	رنم
	ريض	ريش	ريس	ريح	ريث	ريب	روي	روم	رول	روق	روغ	روع	روض	روز	رود	روح
										ريي	رين	ريم	ريل	ريق	ريف	ريع
زجو	زجل	زجر	زجج	زبي	زبن	زبل	زبق	زبط	زبر	زبد	زبب	زأن	زأم	زأق	زأط	زأر
زعط	زعر	زعج	زري	زرق	زرف	زرع	زرر	زرد	زرب	زخم	زخر	زخخ	زحم	زحل	زحف	زحر
زقو	زقم	زقل	زقق	زفن	زفف	زفر	زفت	زغل	زغط	زغر	زغد	زغب	زعم	زعل	زعق	زعف
زمع	زمط	زمر	زمت	زلم	زلل	زلق	زلف	زلع	زلط	زلج	زكي	زكو	زكن	زكم	زكر	زكب
زهو	زهم	زهق	زهف	زهر	زهد	زني	زنن	زنق	زنر	زند	زنخ	زنج	زنأ	زمن	زمم	زمل
زيف	زيغ	زيز	زير	زيد	زيح	زيت	زوي	زون	زوم	زول	زوق	زوغ	زور	زود	زوح	زوج
										زيي	زين	زيل	زيق			
سبه	سبل	سبك	سبق	سبغ	سبع	سبط	سبس	سبر	سبخ	سبح	سبت	سبب	سبأ	سأم	سأل	سأر
سجي	سجو	سجن	سجم	سجل	سجق	سجف	سجع	سجس	سجر	سجد	سجح	سته	ستف	ستر	ستت	سبي
سخم	سخل	سخف	سخط	سخر	سخد	سحي	سحن	سحم	سحل	سحق	سحف	سحر	سحح	سحج	سحت	سحب
سرح	سرج	سرب	سذج	سذب	سدي	سدن	سدم	سدل	سدف	سدس	سدر	سدد	سدب	سخي	سخو	سخن
سعد	سطو	سطل	سطع	سطر	سطح	سطب	سري	سرو	سرم	سرق	سرف	سرع	سرط	سرر	سرد	سرخ
سفن	سفل	سفك	سفق	سفف	سفع	سفط	سفر	سفد	سفح	سغب	سعي	سعن	سعل	سعف	سعط	سعر

سكن	سكك	سكف	سكع	سكر	سكت	سكب	سقي	سقو	سقم	سقل	سقف	سقع	سقط	سقر	سفي	سفه
سمج	سمت	سلي	سلو	سلم	سلل	سلك	سلق	سلف	سلع	سلط	سلس	سلخ	سلح	سلت	سلب	سلأ
سنر	سند	سنخ	سنح	سنج	سمي	سمو	سمن	سمم	سمل	سمك	سمق	سمع	سمط	سمر	سمد	سمح
سود	سوخ	سوح	سوج	سوأ	سهو	سهم	سهل	سهف	سهر	سهد	سهب	سنو	سنه	سنن	سنم	سنط
سيد	سيخ	سيح	سيج	سيب	سيأ	سوي	سوم	سول	سوك	سوق	سوف	سوغ	سوع	سوط	سوس	سور
													سيل	سيف	سيس	سير
شتت	شبو	شبه	شبن	شبل	شبك	شبق	شبع	شبط	شبر	شبح	شبث	شبب	شأو	شأن	شأم	شأب
شحط	شحر	شحذ	شحح	شحت	شحب	شجي	شجو	شجن	شجع	شجر	شجج	شجب	شتو	شتم	شتل	شتر
شذب	شدو	شده	شدن	شدق	شدف	شدر	شدد	شدخ	شخم	شخط	شخص	شخر	شخخ	شخب	شحن	شحم
شرم	شرك	شرق	شرف	شرع	شرط	شرش	شرس	شرر	شرد	شرخ	شرح	شرج	شرب	شذو	شذر	شذذ
شعب	شظي	شظف	شطن	شطف	شطط	شطر	شطح	شطب	شطأ	شصص	شصر	شسع	شزر	شري	شرو	شره
شفف	شفع	شفط	شفر	شفت	شغل	شغف	شغر	شغب	شعو	شعن	شعل	شعف	شعع	شعط	شعر	شعث
شكل	شكك	شكس	شكر	شكد	شقي	شقو	شقل	شقق	شقف	شقر	شقح	شفي	شفو	شفه	شفن	شفق
شمع	شمط	شمس	شمر	شمخ	شمت	شلو	شلل	شلق	شلف	شلح	شلت	شلب	شكي	شكو	شكه	شكم
شهل	شهق	شهر	شهد	شهب	شنن	شنق	شنف	شنع	شنط	شنر	شنخ	شنج	شنب	شنأ	شمم	شمل
شوه	شون	شوم	شول	شوك	شوق	شوف	شوظ	شوط	شوش	شور	شوح	شوب	شهي	شهو	شهن	شهم
			شين	شيم	شيل	شيق	شيف	شيع	شيط	شيد	شيخ	شيح	شيت	شيب	شيأ	شوي

صحل	صحف	صحر	صحح	صحب	صتم	صبي	صبو	صبن	صبغ	صبع	صبر	صبح	صبب	صبأ	صأي	صأب
صرح	صرب	صدي	صدم	صدق	صدف	صدغ	صدع	صدر	صدد	صدح	صدأ	صخر	صخب	صحي	صحو	صحن
صغو	صغر	صعل	صعق	صعر	صعد	صعب	صطل	صطب	صري	صرم	صرف	صرع	صرط	صرر	صرد	صرخ
صلت	صلب	صكك	صقل	صقع	صقر	صقب	صفي	صفو	صفن	صفق	صفف	صفع	صفر	صفد	صفح	صغي
صمم	صمل	صمغ	صمد	صمخ	صمت	صلي	صلو	صلن	صلل	صلف	صلع	صلص	صلد	صلخ	صلح	صلج
صوح	صوج	صوت	صوب	صهو	صهل	صهر	صهد	صهب	صنو	صنن	صنم	صنف	صنع	صنر	صنج	صمي
صيف	صيغ	صيع	صيص	صير	صيد	صيح	صيب	صون	صوم	صول	صوف	صوغ	صوع	صوص	صور	صوخ
															صين	صيم
ضخخ	ضحي	ضحو	ضحل	ضحك	ضجع	ضجر	ضجج	ضبن	ضبع	ضبط	ضبس	ضبر	ضبح	ضبب	ضأن	ضأل
ضفف	ضفر	ضغن	ضغط	ضغث	ضعف	ضرو	ضرم	ضرع	ضرط	ضرس	ضرر	ضرح	ضرج	ضرب	ضدد	ضخم
ضوأ	ضهي	ضهل	ضهر	ضهد	ضني	ضنن	ضنك	ضمن	ضمم	ضمر	ضمد	ضمخ	ضمج	ضلل	ضلع	ضفو
								ضيم	ضيق	ضيف	ضيع	ضير	ضوي	ضوع	ضور	ضوج
طرب	طرأ	طخي	طخر	طحن	طحل	طحر	طجن	طبي	طبن	طبل	طبق	طبع	طبر	طبخ	طبب	طأس
طعن	طعم	طشت	طست	طزن	طزج	طري	طرو	طرم	طرق	طرف	طرش	طرس	طرز	طرر	طرد	طرح
طقم	طقق	طقس	طفي	طفو	طفل	طفق	طفف	طفش	طفر	طفح	طفأ	طغي	طغو	طغم	طغر	طغت
طمن	طمم	طمع	طمس	طمر	طمح	طمث	طلي	طلو	طلم	طلل	طلق	طلع	طلس	طلح	طلب	طقي
طوش	طوس	طور	طود	طوح	طوب	طهي	طهو	طهم	طهق	طهر	طنن	طنف	طنج	طنب	طمي	طمو

طوع	طوف	طوق	طول	طوي	طيب	طيح	طير	طيش	طيع	طيف	طين					
ظبي	ظرر	ظرف	ظعن	ظفر	ظلع	ظلف	ظلل	ظلم	ظمأ	ظنب	ظنن	ظهر				
عباً	عبب	عبث	عبد	عبر	عبس	عبط	عبق	عبك	عبل	عبو	عبي	عتب	عتد	عتر	عتق	عتك
عتل	عتم	عته	عتو	عتي	عثث	عثر	عثم	عثن	عثو	عثي	عجب	عجج	عجر	عجز	عجف	عجل
عجم	عجن	عجو	عدد	عدس	عدل	عدم	عدن	عدو	عذب	عذر	عذق	عذل	عذو	عذي	عرب	عرج
عرد	عرر	عرس	عرش	عرص	عرض	عرف	عرق	عرك	عرم	عرن	عرو	عري	عزب	عزر	عزز	عزف
عزق	عزل	عزم	عزو	عزي	عسب	عسر	عسس	عسف	عسل	عسو	عسي	عشب	عشر	عشش	عشق	عشم
عشو	عشي	عصب	عصد	عصر	عصص	عصف	عصل	عصم	عصو	عصي	عضب	عضد	عضض	عضل	عضه	عضو
عطب	عطر	عطس	عطش	عطف	عطل	عطن	عطي	عظل	عظم	عظي	عفر	عفس	عفش	عفص	عفف	عفن
عفو	عقب	عقد	عقر	عقص	عقف	عقق	عقل	عقم	عكد	عكر	عكز	عكس	عكش	عكف	عكك	عكم
علب	علج	علف	علق	علك	علل	علم	علن	علو	علي	عمد	عمر	عمش	عمص	عمق	عمل	عمم
عمن	عمه	عمي	عنب	عنت	عنج	عند	عنز	عنس	عنف	عنق	عنن	عنو	عني	عهد	عهر	عهل
عهن	عوج	عود	عوذ	عور	عوز	عوص	عوض	عوف	عوق	عول	عوم	عون	عوه	عوي	عيب	عيث
عيد	عير	عيس	عيش	عيط	عيف	عيل	عين	عيه	عيي							
غبب	غبر	غبس	غبش	غبط	غبن	غبو	غبي	غتت	غثث	غثي	غجر	غدد	غدر	غدف	غدق	غدن
غدو	غدي	غذذ	غذو	غرب	غرد	غرر	غرز	غرس	غرش	غرض	غرف	غرق	غرل	غرم	غرن	غرو
غري	غزر	غزز	غزل	غزو	غسس	غسق	غسل	غسن	غشش	غشم	غشو	غشي	غصب	غصص	غصن	غضب

غلب	غقق	غفي	غفو	غفل	غفق	غفف	غفر	غطو	غطط	غطش	غطس	غطر	غضو	غضن	غضض	غضر
غمط	غمض	غمص	غمش	غمس	غمز	غمر	غمد	غلي	غلو	غلم	غلل	غلق	غلف	غلظ	غلط	غلس
غول	غوغ	غوط	غوص	غوش	غوز	غور	غوث	غواً	غني	غنن	غنم	غنص	غنج	غمي	غمم	غمق
					غيي	غين	غيم	غيل	غيظ	غيط	غيض	غير	غيد	غيث	غيب	غوي
فتي	فتو	فتن	فتل	فتك	فتق	فتش	فتر	فتخ	فتح	فتت	فتأ	فأل	فأس	فأر	فأد	فأت
فخر	فخذ	فخخ	فخت	فحو	فحم	فحل	فحص	فحش	فحح	فجو	فجل	فجع	فجر	فجج	فجأ	فثأ
فرش	فرس	فرز	فرر	فرد	فرخ	فرح	فرج	فرأ	فذذ	فدي	فدن	فدم	فدر	فدخ	فدح	فخم
فسخ	فسح	فزع	فزز	فزر	فري	فرو	فره	فرن	فرم	فرك	فرق	فرغ	فرع	فرط	فرض	فرص
فصي	فصم	فصل	فصص	فصد	فصح	فشو	فشل	فشك	فشش	فشر	فشخ	فسو	فسل	فسق	فسر	فسد
فغر	فعي	فعو	فعم	فعل	فظع	فظظ	فطن	فطم	فطس	فطر	فطح	فضي	فضو	فضل	فضض	فضح
فكن	فكك	فكش	فكر	فقه	فقم	فقع	فقط	فقص	فقش	فقس	فقر	فقد	فقح	فقأ	فغو	فغم
فمم	فلي	فلو	فلن	فلم	فلل	فلك	فلق	فلع	فلط	فلس	فلز	فلذ	فلح	فلج	فلت	فكه
فود	فوح	فوج	فوت	فهه	فهم	فهق	فهد	فني	فنن	فنك	فنق	فنط	فنس	فنر	فند	فنخ
فيل	فيف	فيظ	فيض	فيش	فيد	فيح	فيأ	فوه	فول	فوق	فوف	فوع	فوط	فوض	فوز	فور
																فين
قتر	قتد	قتت	قتب	قبو	قبن	قبل	قبع	قبط	قبض	قبص	قبس	قبر	قبح	قبج	قبب	قأد
قدو	قدم	قدس	قدر	قدد	قدح	قحو	قحم	قحل	قحف	قحط	قحح	قحب	قثث	قثأ	قتم	قتل

قرص	قرش	قرس	قرر	قرد	قرح	قرج	قرت	قرب	قرأ	قذي	قذل	قذف	قذع	قذر	قذذ	قدي
قسح	قزن	قزم	قزل	قزع	قززز	قزح	قري	قرو	قرن	قرم	قرق	قرف	قرع	قرظ	قرط	قرض
قصر	قصد	قصج	قصب	قشل	قشف	قشع	قشط	قشش	قشر	قشد	قشب	قسو	قسم	قسط	قسس	قسر
قطط	قطر	قطب	قضي	قضو	قضم	قضف	قضع	قضض	قضب	قصي	قصو	قصم	قصل	قصف	قصع	قصص
قفل	قفف	قفع	قفص	قفش	قفز	قفر	قعي	قعس	قعر	قعد	قطو	قطن	قطم	قطل	قطف	قطع
قمح	قمأ	قلي	قلو	قلم	قلل	قلق	قلف	قلع	قلط	قلص	قلش	قلس	قلد	قلح	قلب	قفو
قنط	قنص	قند	قنج	قنت	قنب	قنأ	قمن	قمم	قمل	قمع	قمط	قمص	قمش	قمس	قمز	قمر
قوص	قوش	قوس	قور	قود	قوح	قوت	قوب	قهي	قهو	قهر	قني	قنو	قنن	قنم	قنل	قنع
قيظ	قيض	قيش	قيس	قير	قيد	قيح	قيأ	قوي	قوه	قون	قوم	قول	قوق	قوع	قوط	قوض
												قيي	قين	قيم	قيل	قيف
كتع	كتت	كتب	كبي	كبو	كبن	كبل	كبش	كبس	كبر	كبد	كبح	كبت	كبب	كأس	كأد	كأب
كدش	كدس	كدر	كدد	كدح	كخي	كحل	كحح	كحت	كثف	كثر	كثث	كثب	كتن	كتم	كتل	كتف
كره	كرم	كرك	كرع	كرط	كرش	كرس	كرز	كرر	كرد	كرح	كرج	كرث	كرب	كذب	كدي	كدم
كشط	كشش	كشر	كشح	كسي	كسو	كسم	كسل	كسف	كسع	كسر	كسد	كسح	كسب	كزز	كري	كرو
كفر	كفخ	كفح	كفت	كفأ	كغط	كغذ	كغد	كعم	كعك	كعب	كظم	كظظ	كظر	كضض	كشك	كشف
كلو	كلن	كلم	كلل	كلك	كلف	كلس	كلد	كلح	كلت	كلب	كلأ	كفي	كفن	كفل	كفف	كفس
كنر	كند	كنب	كمي	كمه	كمن	كمم	كمل	كمع	كمش	كمر	كمد	كمخ	كمح	كمت	كمأ	كلي
كود	كوخ	كوب	كهي	كهن	كهم	كهل	كهف	كني	كنو	كنه	كنن	كنف	كنع	كنش	كنس	كنز

كيي	كين	كيل	كيف	كيس	كير	كيد	كوي	كون	كوم	كوك	كوف	كوع	كوش	كوس	كوز	كور
لثم	لثغ	لتت	لبي	لبو	لبن	لبك	لبق	لبط	لبس	لبد	لبخ	لبث	لبب	لبأ	لأم	لأك
لحو	لحن	لحم	لحق	لحف	لحظ	لحس	لحد	لحح	لحج	لحب	لجن	لجم	لجج	لجب	لجأ	لثو
لسع	لزم	لزق	لزز	لزج	لزب	لذي	لذق	لذع	لذذ	لدن	لدغ	لدد	لخن	لخم	لخص	لحي
لعن	لعق	لعس	لعج	لعب	لظي	لطم	لطف	لطع	لطش	لطس	لطخ	لضم	لصق	لصص	لشي	لسن
لقط	لقس	لقح	لقب	لفو	لفق	لفف	لفع	لفظ	لفح	لفت	لغو	لغم	لغط	لغز	لغد	لغب
لمص	لمس	لمز	لمح	لمج	لكن	لكم	لكك	لكع	لكز	لكأ	لقي	لقو	لقن	لقم	لقف	لقع
لوح	لوج	لوث	لوب	لهي	لهو	لهم	لهق	لهف	لهط	لهد	لهج	لهث	لهب	لمم	لمع	لمظ
ليل	ليق	ليف	ليس	ليث	لوي	لون	لوم	لوك	لوق	لوف	لوع	لوط	لوص	لوس	لوز	لوذ
														ليي	لين	ليم
مجن	مجل	مجس	مجر	مجد	مجج	مثن	مثل	مثث	متن	متك	متع	متر	متح	متت	مأن	مأق
مدر	مدد	مدح	مخل	مخط	مخض	مخر	مخخ	محي	محو	محن	محل	محك	محق	محض	محص	محح
مرط	مرض	مرش	مرس	مرر	مرد	مرخ	مرح	مرج	مرث	مرأ	مذل	مذق	مذر	مدي	مدو	مدن
مسد	مسخ	مسح	مزي	مزن	مزق	مزع	مزز	مزر	مزح	مزج	مري	مرو	مرن	مرق	مرغ	مرع
مضر	مصل	مصص	مصر	مشي	مشو	مشك	مشق	مشط	مشش	مشح	مشج	مسي	مسو	مسك	مسس	مسر
معك	معط	معض	معس	معز	معر	معد	معج	مطي	مطو	مطل	مطق	مطط	مطر	مضي	مضغ	مضض
مكن	مكك	مكس	مكر	مكث	مقل	مقع	مقت	مغن	مغل	مغط	مغص	مغر	مغث	معي	معو	معن

منع	منح	منأ	ملي	ملو	ملم	ملل	ملك	ملق	ملط	ملص	ملس	ملد	ملخ	ملح	ملج	ملأ
مول	مور	موج	موت	موأ	مهي	مهو	مهن	مهل	مهك	مهق	مهر	مهد	مهج	مني	منو	منن
					مين	ميل	ميع	ميط	ميس	ميز	مير	ميد	ميح	ميت	موه	مون
نبق	نبغ	نبع	نبط	نبض	نبش	نبس	نبز	نبر	نبذ	نبح	نبج	نبت	نبب	نبأ	نأي	نأم
نجذ	نجد	نجح	نجب	نثر	نتن	نتف	نتع	نتش	نتر	نتح	نتج	نتأ	نبو	نبه	نبل	نبك
نحم	نحل	نحف	نحس	نحز	نحر	نحت	نحب	نجو	نجم	نجل	نجف	نجع	نجش	نجس	نجز	نجر
ندل	ندف	ندس	ندر	ندد	ندح	ندب	نخو	نخم	نخل	نخع	نخس	نخز	نخر	نخخ	نخب	نحو
نزه	نزل	نزك	نزق	نزف	نزغ	نزع	نزز	نزر	نزح	نرد	نذل	نذر	ندي	ندو	نده	ندم
نشج	نشب	نشأ	نسي	نسو	نسم	نسل	نسك	نسق	نسف	نسغ	نسر	نسخ	نسج	نسب	نسأ	نزو
نصع	نصص	نصر	نصح	نصت	نصب	نشو	نشن	نشل	نشق	نشف	نشع	نشط	نشش	نشز	نشر	نشد
نطط	نطس	نطر	نطح	نضو	نضل	نضف	نضض	نضر	نضد	نضح	نضج	نضب	نصو	نصم	نصل	نصف
نعل	نعق	نعظ	نعش	نعس	نعر	نعج	نعت	نعب	نظم	نظل	نظف	نظر	نطل	نطق	نطف	نطع
نفر	نفذ	نفد	نفخ	نفح	نفج	نفث	نغي	نغو	نغم	نغل	نغص	نغش	نغز	نغب	نعي	نعم
نقس	نقز	نقر	نقذ	نقد	نقح	نقب	نفي	نفو	نفل	نفق	نفف	نفع	نفط	نفض	نفش	نفس
نكح	نكث	نكت	نكب	نكأ	نقي	نقو	نقه	نقم	نقل	نقق	نقف	نقع	نقط	نقض	نقص	نقش
نمم	نمل	نمق	نمط	نمش	نمس	نمر	نكي	نكه	نكل	نكف	نكص	نكش	نكس	نكز	نكر	نكد
نوب	نوأ	نهي	نهو	نهم	نهل	نهك	نهق	نهض	نهش	نهز	نهر	نهد	نهج	نهب	نمي	نمو
نوو	نوه	نون	نوم	نول	نوق	نوف	نوع	نوط	نوص	نوش	نوس	نور	نود	نوخ	نوح	نوت

هجر	هجد	هجج	هجأ	هتن	هتم	هتك	هتف	هتر	هبو	هبل	هبط	هبش	هبر	هبت	هبب	هأل
هدي	هدن	هدم	هدل	هدف	هدر	هدد	هدج	هدب	هدأ	هجو	هجن	هجم	هجل	هجع	هجص	هجس
هزأ	هري	هرو	هرم	هرق	هرف	هرع	هرش	هرس	هرر	هرج	هرب	هرأ	هذي	هذل	هذر	هذب
هفت	هطل	هطع	هضم	هضض	هضب	هصص	هصر	هشم	هشش	هسس	هزم	هزل	هزع	هزز	هزر	هزج
همس	همز	همر	همد	همج	هلن	هلم	هلل	هلك	هلع	هلس	هلب	هكم	هكع	هكر	هفو	هفف
هوج	هوت	هوب	هني	هنو	هنه	هنن	هنم	هنف	هند	هنأ	همو	همم	همل	همك	همع	همش
هيض	هيش	هير	هيج	هيت	هيب	هيأ	هوي	هوو	هون	هوم	هول	هوع	هوش	هوس	هور	هود
												هين	هيم	هيل	هيف	هيط
وثب	وثأ	وتي	وتن	وتر	وتد	وبه	وبل	وبق	وبش	وبر	وبخ	وبأ	وأم	وأل	وأر	وأد
وحد	وجه	وجن	وجم	وجل	وجق	وجف	وجع	وجس	وجز	وجر	وجد	وجب	وثن	وثل	وثق	وثر
وذر	ودي	ودك	ودق	ودع	ودر	ودد	ودج	وخي	وخم	وخط	وخز	وحي	وحم	وحل	وحف	وحش
وزز	وزر	وزب	وري	ورن	ورم	ورل	ورك	ورق	ورف	ورع	ورط	ورش	ورس	ورد	ورث	ورب
وشر	وشح	وشج	وشب	وسي	وسن	وسم	وسل	وسق	وسع	وسط	وسد	وسخ	وزي	وزن	وزل	وزع
وضر	وضح	وضب	وضأ	وصي	وصم	وصل	وصف	وصد	وصب	وشي	وشن	وشم	وشل	وشك	وشق	وشع
وعر	وعد	وعث	وعب	وظف	وظب	وطي	وطن	وطف	وطش	وطس	وطر	وطد	وطب	وطأ	وضم	وضع
وفق	وفع	وفض	وفز	وفر	وفد	وغي	وغل	وغر	وغد	وعي	وعل	وعك	وعق	وعظ	وعس	وعز

وكب	وكأ	وقي	وقن	وقل	وقق	وقف	وقع	وقظ	وقص	وقر	وقذ	وقد	وقح	وقت	وقب	وفي
ولف	ولغ	ولع	ولط	ولس	ولد	ولج	وكي	وكن	وكم	وكل	وكف	وكع	وكس	وكز	وكر	وكد
وهل	وهق	وهر	وهد	وهج	وهب	وني	ونن	ومق	ومض	ومس	ومد	ومأ	ولي	ولو	وله	ولم
												ويل	ويب	وهي	وهن	وهم
يقظ	يقت	يفع	يفخ	يسن	يسر	يزب	يرق	يرع	يخن	يخت	يحر	يثق	يتم	يبس	يبب	يأس
											يوم	يود	ينع	يمن	يمم	يقن

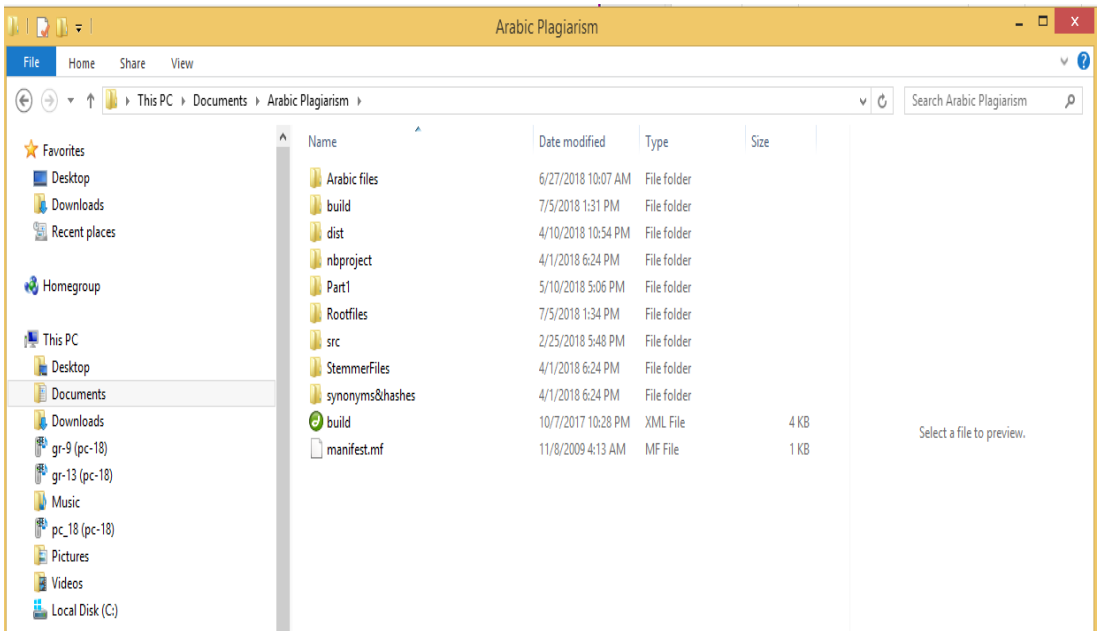
APPENDIX C

CORPUS COLLECTION

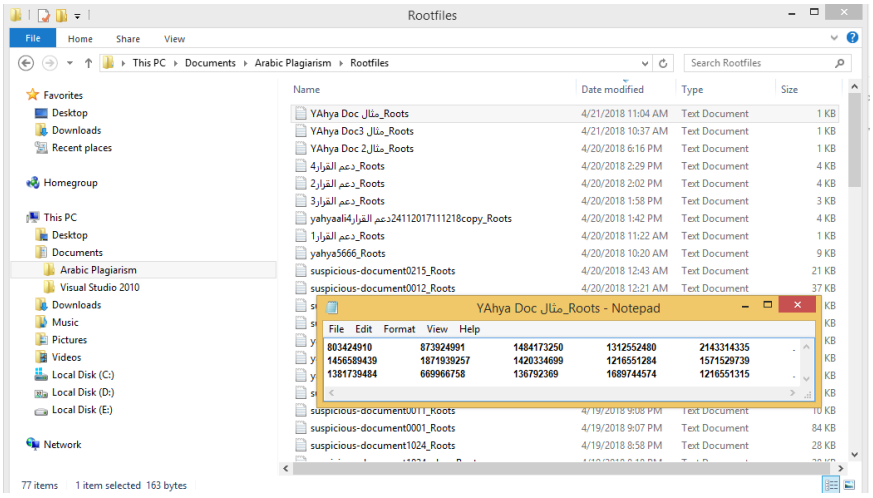
Doc Name	Doc Author	Link
Dataset1	InAraPlagDet-20-06-2015	Create your own country blog http://misc-umc.org/AraPlagDet/?i=1#datasets
Dataset2	InAraPlagDet-20-06-201٥	Islamic book http://misc-umc.org/AraPlagDet/?i=1#datasets
Dataset3	InAraPlagDet-20-06-201٢	Corpus of Classical Arabic http://misc-umc.org/AraPlagDet/?i=1#datasets
Dataset4	wikipedia.org	https://ar.wikipedia.org/wiki/نظام_دعم_قرار

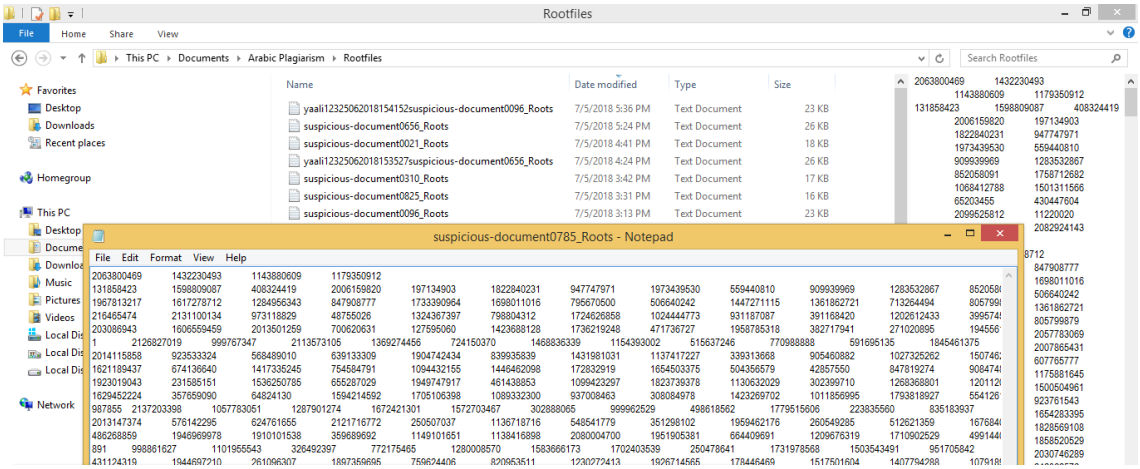
APPENDIX E

ADPDM Files and Result Details

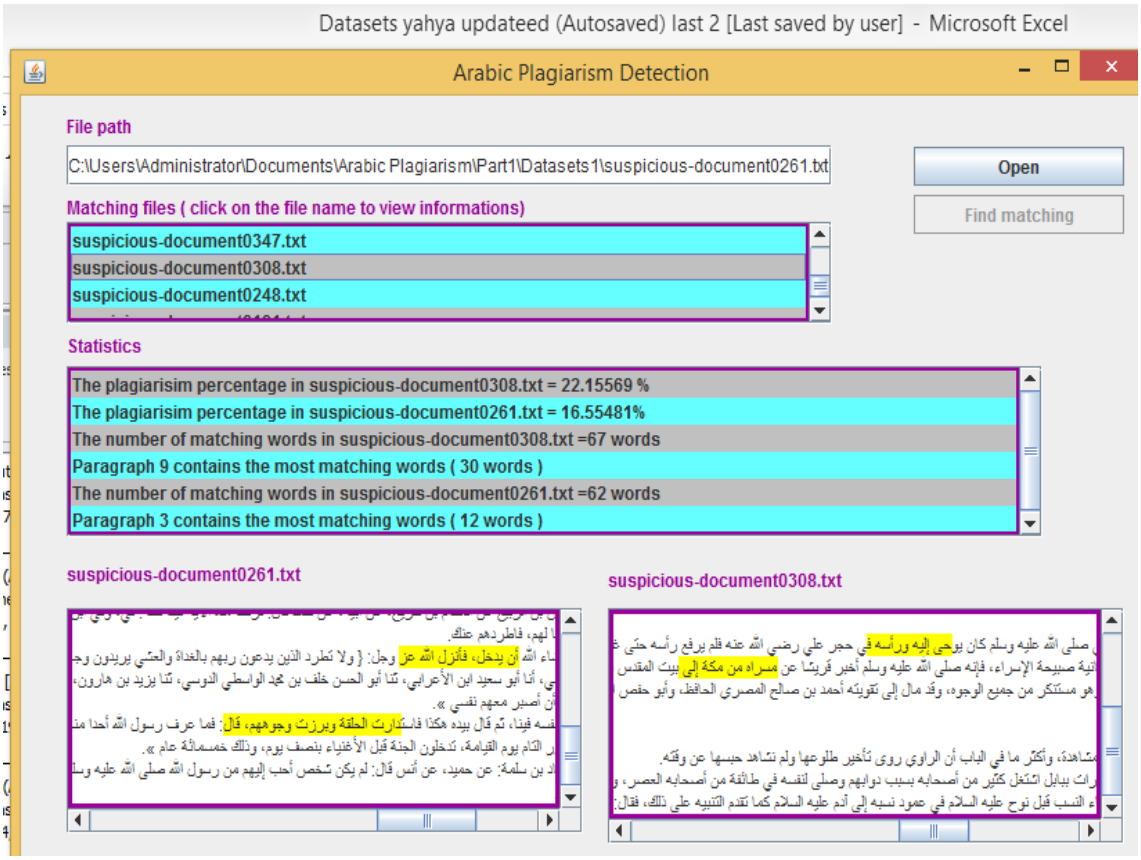


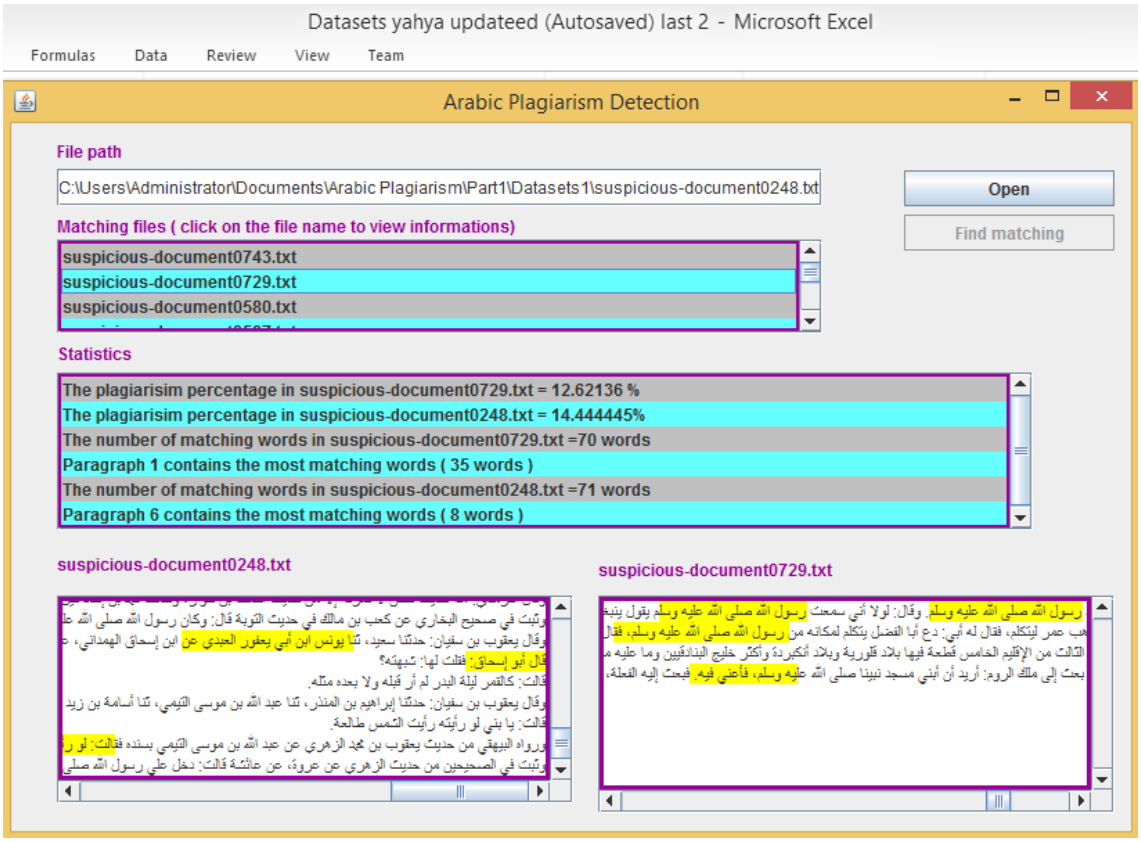
Example of Fingerprinting Documents base on 3-gram hash



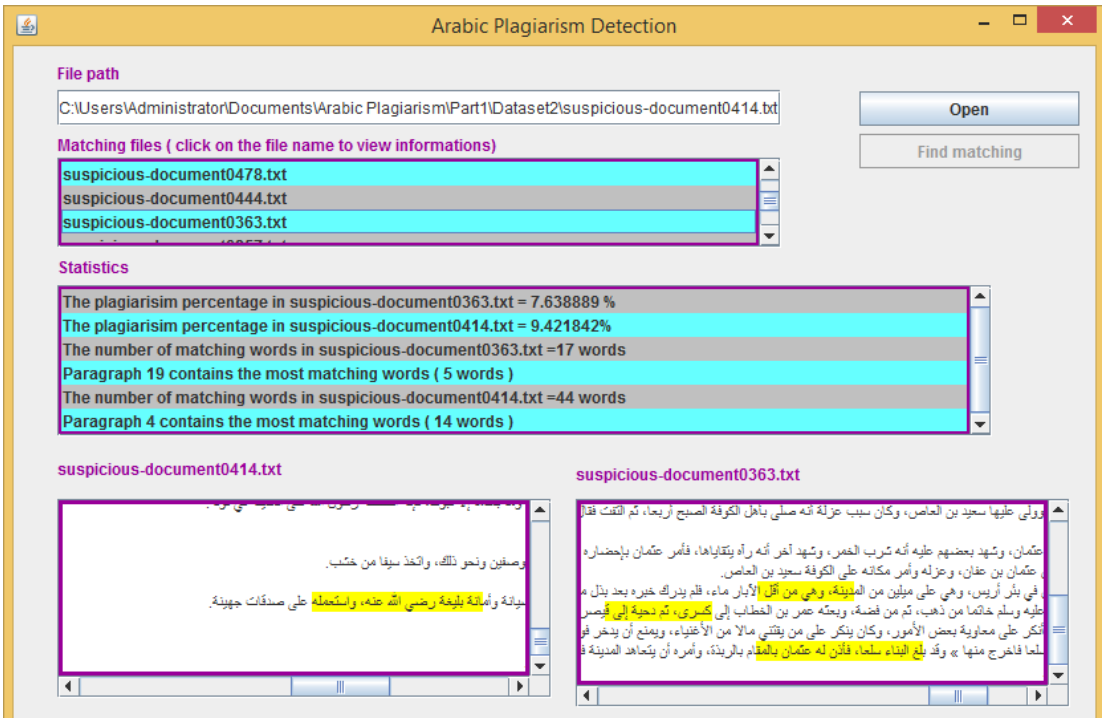


Experimental Result Dataset1

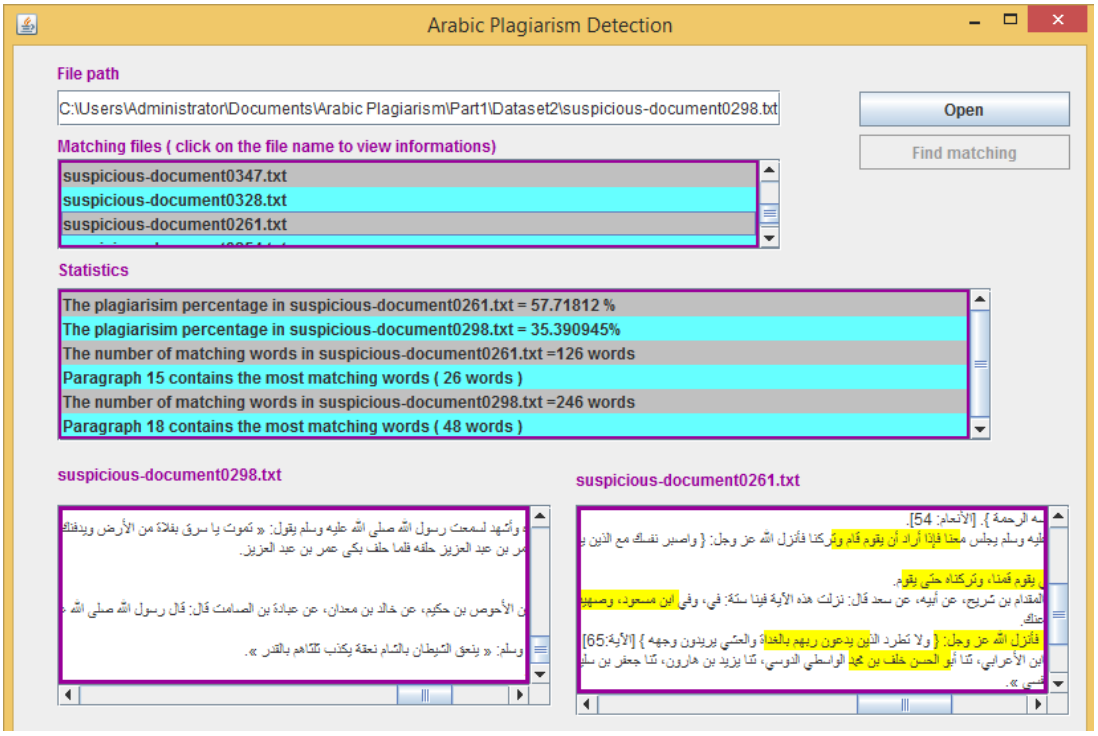


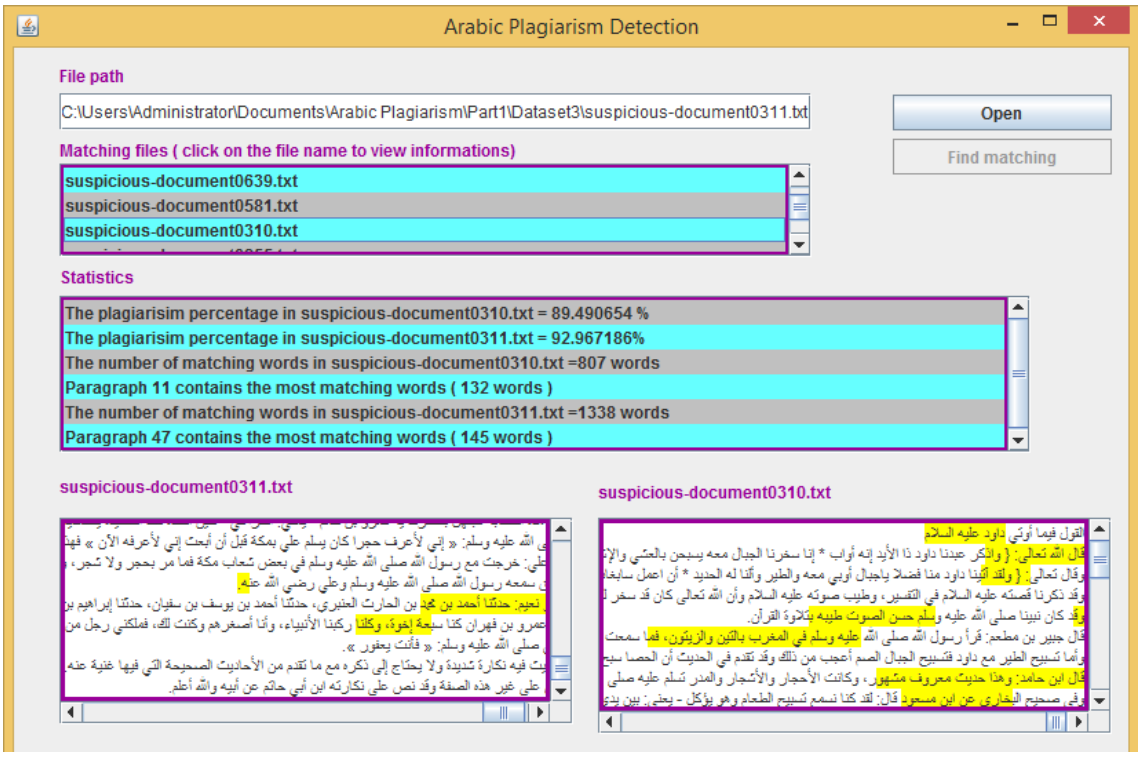


experimental Result Dataset2

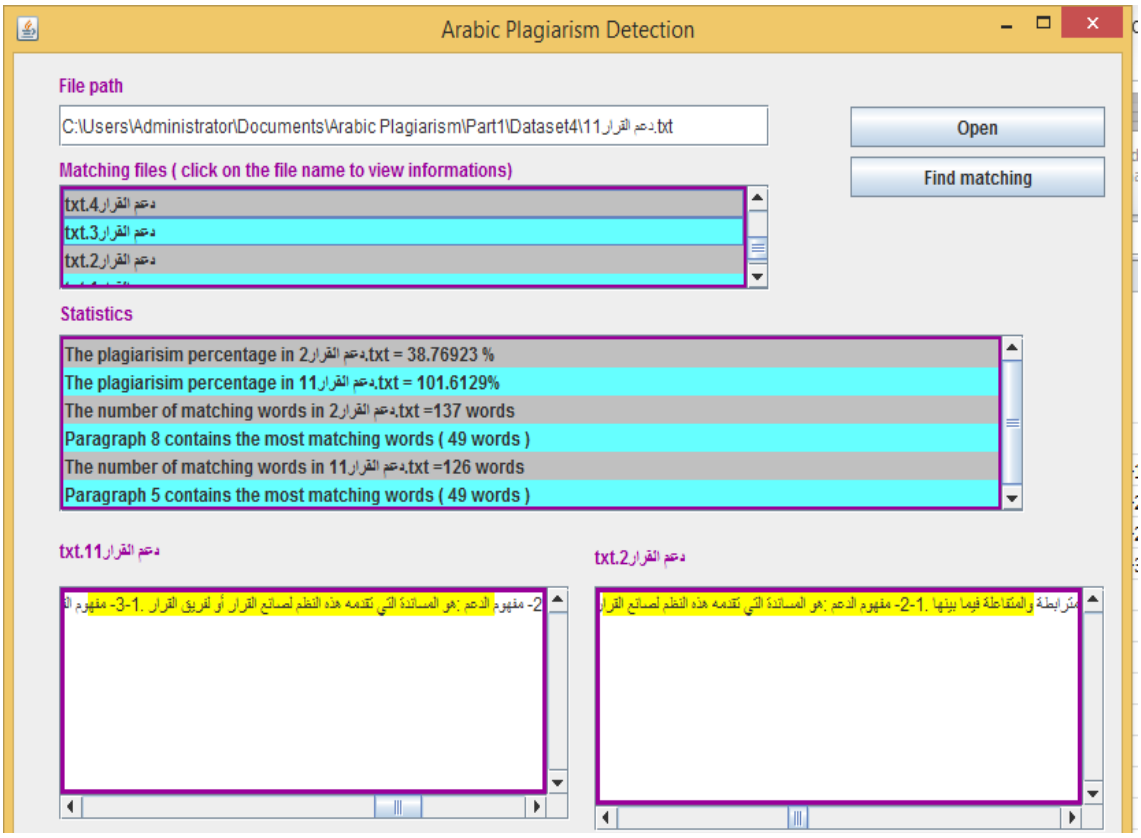


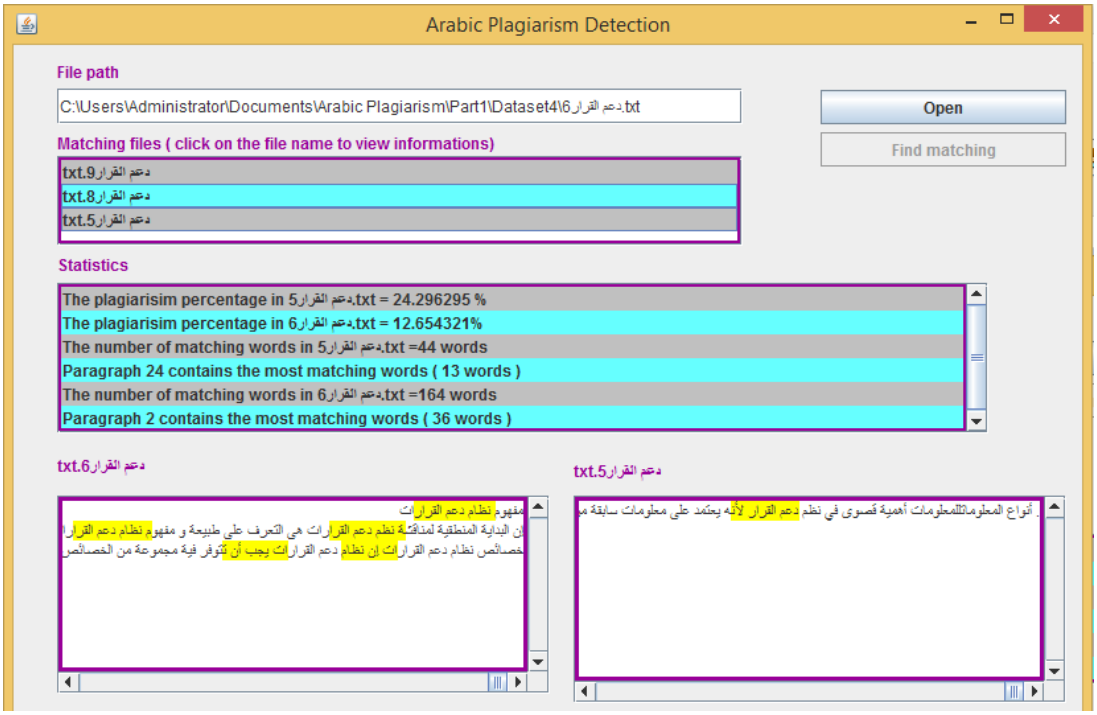
Experimental Result Dataset3

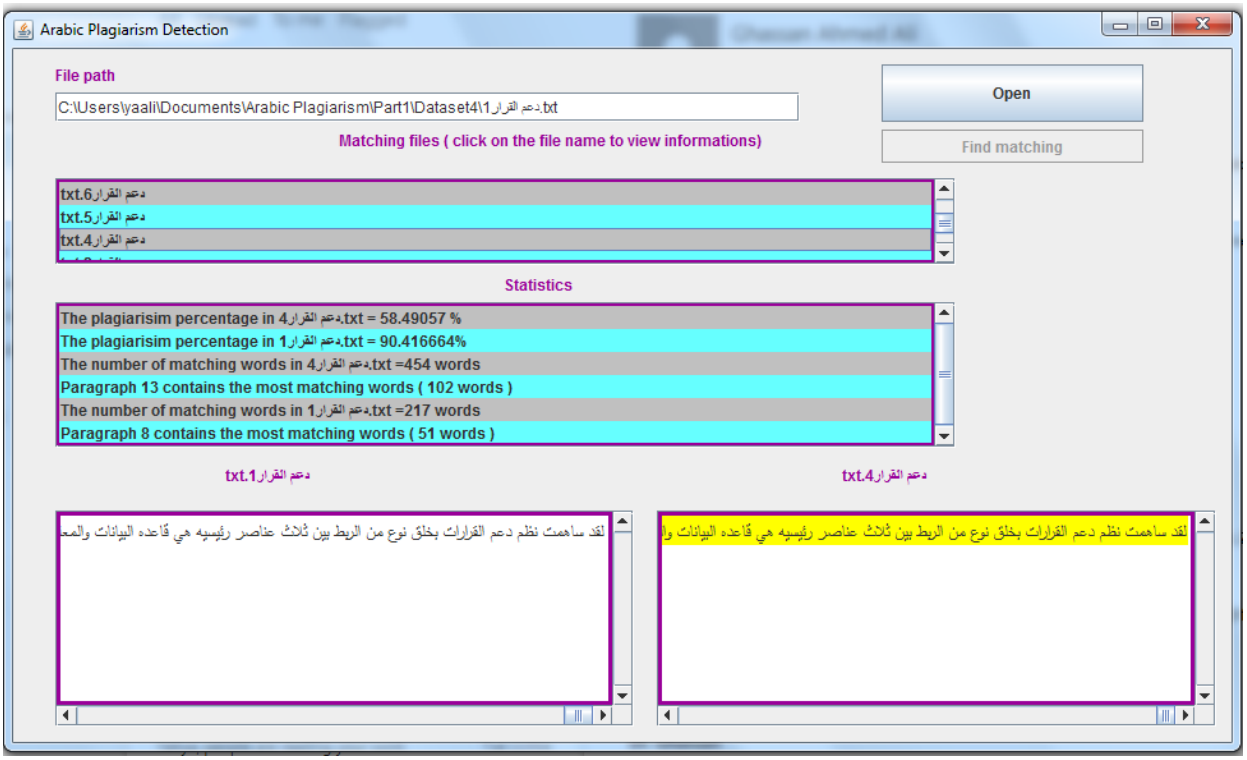


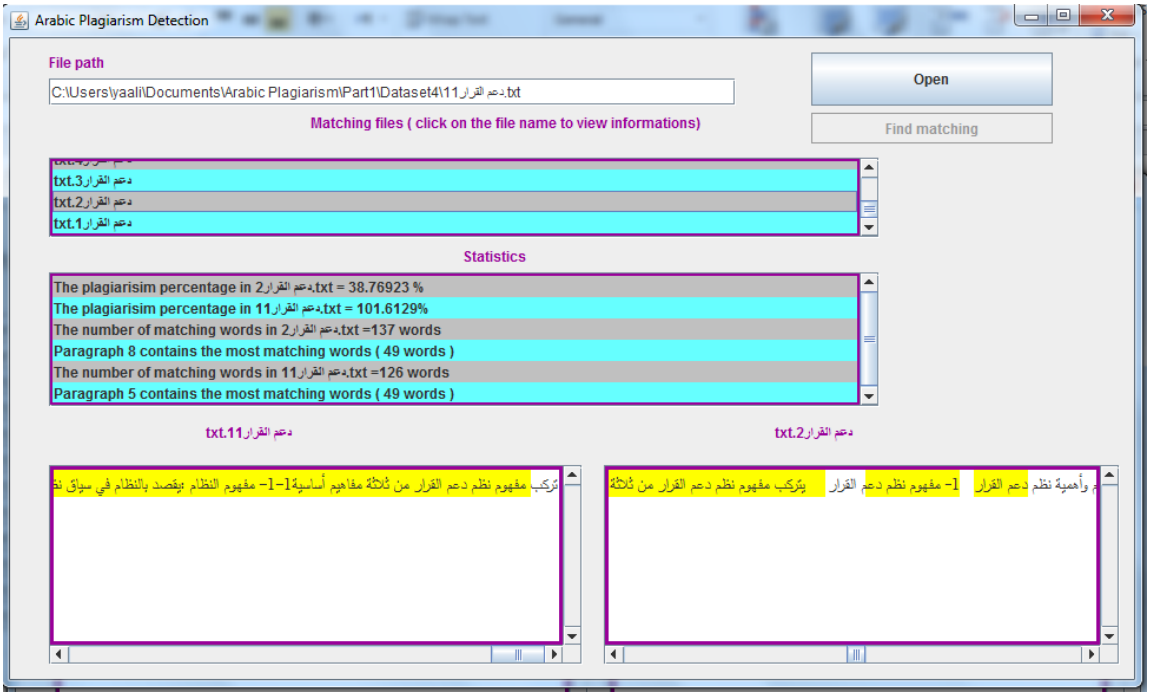


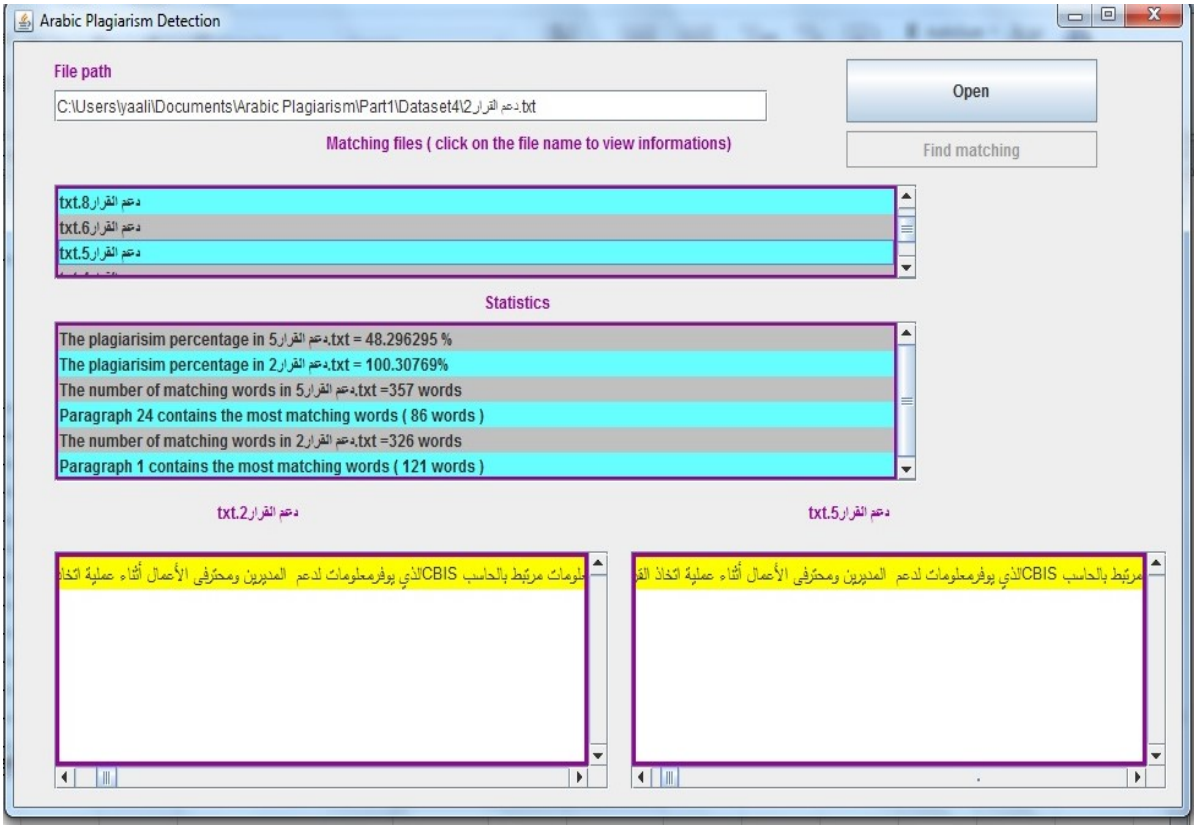
Epremental Result Dataset4






Arabic Plagiarism Detection
File path
C:\Users\yaali\Documents\Arabic Plagiarism\Part1\Dataset4\1دعم القرار.txt
Open
Find matching
Matching files (click on the file name to view informations)
دعم القرار6.txt
دعم القرار5.txt
دعم القرار4.txt
Statistics
The plagiarisim percentage in 4دعم القرار.txt = 58.49057 %
The plagiarisim percentage in 1دعم القرار.txt = 90.416664%
The number of matching words in 4دعم القرار.txt =454 words
Paragraph 13 contains the most matching words (102 words)
The number of matching words in 1دعم القرار.txt =217 words
Paragraph 8 contains the most matching words (51 words)
دعم القرار1.txt
دعم القرار4.txt
لقد ساهمت نظم دعم القرارات بخلق نوع من الربط بين ثلاث عناصر رئيسيه هي قاعده البيانات والمعـ
لقد ساهمت نظم دعم القرارات بخلق نوع من الربط بين ثلاث عناصر رئيسيه هي قاعده البيانات وا


Arabic Plagiarism Detection
File path
C:\Users\yaali\Documents\Arabic Plagiarism\Part1\Dataset4\11دعم القرار.txt
Open
Find matching
Matching files (click on the file name to view informations)
دعم القرار3.txt
دعم القرار2.txt
دعم القرار1.txt
Statistics
The plagiarisim percentage in 2دعم القرار.txt = 38.76923 %
The plagiarisim percentage in 11دعم القرار.txt = 101.6129%
The number of matching words in 2دعم القرار.txt =137 words
Paragraph 8 contains the most matching words (49 words)
The number of matching words in 11دعم القرار.txt =126 words
Paragraph 5 contains the most matching words (49 words)
دعم القرار11.txt
دعم القرار2.txt
تركب مفهوم نظم دعم القرار من ثلاثة مفاهيم أساسية1- 1- مفهوم النظام :يقصد بالنظام في سياق نظ
يتركب مفهوم نظم دعم القرار من ثلاثة 1- مفهوم نظم دعم القرار وأهمية نظم دعم القرار


Arabic Plagiarism Detection
File path
C:\Users\yaali\Documents\Arabic Plagiarism\Part1\Dataset4\2.دعم القرار.txt
Open
Find matching
Matching files (click on the file name to view informations)
دعم القرار8.txt
دعم القرار6.txt
دعم القرار5.txt
Statistics
The plagiarisim percentage in 5.دعم القرار.txt = 48.296295 %
The plagiarisim percentage in 2.دعم القرار.txt = 100.30769%
The number of matching words in 5.دعم القرار.txt =357 words
Paragraph 24 contains the most matching words (86 words)
The number of matching words in 2.دعم القرار.txt =326 words
Paragraph 1 contains the most matching words (121 words)
دعم القرار2.txt
دعم القرار5.txt
CBISالذي يوفرمعلومات لدعم المديرين ومحترفي الأعمال أثناء عملية اتخاذ
CBISالذي يوفرمعلومات لدعم المديرين ومحترفي الأعمال أثناء عملية اتخاذ

LIST OF PUBLICATION

1 Yahya. A. Abdelrahman , A. Khalid and I. M. Osman," A SURVEY OF PLAGIARISM ETECTION FOR ARABIC DOCUMENTS", INTERNATIONAL JOURNAL OF ADVANCED COMPUTER TECHNOLOGY VOLUME 4, NUMBER 6, (Page no:34-38) (IJACT) December 2015.

2 Yahya. A. Abdelrahman, A. Khalid and I. M. Osman, "A Method For Arabic Documents Plagiarism Detection" International Journal of Computer Science and Information Security(IJCSIS), vol. 15, p. 79, 2017.USA