

## Decision Support System Tool for Arabic Text Recognition

Fatmah Baothman\*, Sarah Alssagaff and Bayan Ashmeel

Information System Department, King Abdul-Aziz University, Jeddah, 21551, Saudi Arabia

\*Corresponding Author: Fatmah Baothman. Email: fbaothman@kau.edu.sa

Received: 20 October 2020; Accepted: 14 December 2020

**Abstract:** The National Center for Education Statistics study reported that 80% of students change their major or institution at least once before getting a degree, which requires a course equivalency process. This error-prone process varies among disciplines, institutions, regions, and countries and requires effort and time. Therefore, this study aims to overcome these issues by developing a decision support tool called TiMELY for automatic Arabic text recognition using artificial intelligence techniques. The developed tool can process a complete document analysis for several course descriptions in multiple file formats, such as Word, Text, Pages, JPEG, GIF, and JPG. We applied a comparative approach in selecting the highest score using three Arabic text extraction algorithms: term frequency-inverse document frequency measure algorithm, Cortical.io tool with Retina Database, and keyword extraction using word co-occurrence algorithm. The data repository consisted of 1000 datasets built from five different faculties at King Abdul-Aziz University and King Faisal University. It was followed by a discussion of the evaluation techniques using precision and recall measurements, which indicated that the keyword extraction using word co-occurrence algorithm scored 90% for the English language and 80% for the Arabic language in terms of the F1 measure that focuses on the linguistic relation between words.

**Keywords:** Text mining; artificial intelligence; Arabic text recognition; course equivalency; decision support system; semantic text processing

### 1 Introduction

A study conducted between 2011 and 2014 by the National Center for Education Statistics (NCES) revealed that nearly one-third of students each year realize that they did not enroll in the right institution or select a major of their interest [1]. NCES also conducted the same study in 2017 and showed that 80% of the enrolled college students changed their major within the first 3 years of their studies [2]. To correct their choices, students have the option of changing their major or transferring to another university and go through the course equivalency process. The college faculty or the institution assigns academic affairs and the responsible equivalency administrators to handle students' transfer. Their primary responsibility is to enforce that the correct equivalency procedures are applied for the completed courses before the student's transfer occurs [3]. On average, the course equivalency procedure takes 14 business days, but it takes months in Saudi universities in some cases due to strict governance and manual practice. For



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

example, when a student fills out the transfer application from college A to college B, the request will be denied by the academic affairs if the courses' descriptions are not attached. Moreover, a new request with all required documents must be initiated for transmission to the responsible equivalency.

This study aims to develop a decision support tool for processing course equivalency; it is based on artificial intelligence (AI) using text mining and recognition addressing the Arabic language challenges. The developed tool analyzes and compares the courses' descriptions by extracting relevant keywords in English or Arabic using text mining and recognition techniques. The work highlights the effect of merging AI techniques (text mining and recognition) with the decision support systems (DSSs) and compares the scores of three algorithms' results. Text mining is an automatic process of converting human language text into meaningful and valuable information or knowledge utilizing machine learning algorithms [4]. Meanwhile, text recognition is the process of detecting, recognizing, and segmenting characters from an input image and converting them to an editable text [5]. The proposed tool takes the educational administrative processes into a new dimension of AI-based service and thus enables students to decide in advance whether to transfer to another university or college by informing them if the courses are accredited in the new college. For administration, the process becomes faster, saves time, and requires less effort. The researchers selected a comparative approach using three techniques for creating and testing to minimize the equivalency process' errors during the implementation. The work produced the first bilingual tool specialized in the course equivalency process using AI techniques of text and recognition. The course equivalency's primary problem is that the process varies among disciplines, institutions, regions, and countries. Furthermore, the process is date and place specific because it depends on the time and place of studying the course. Many prestigious international universities, such as McGill University, Washington University, and Iowa State University, have proposed systems to automate the process. However, the systems' operations are still not fully automated and limited only to the English language. The equivalency process for the Arabic language is essential for many universities in several countries, and the administration at King Abdul-Aziz University (KAU) is keen to improve the current process for making it fully automated. Therefore, one of our created TiMELY tool's main objectives is to support management for electronically equalizing courses completed for transferred students from an accredited university to KAU or even among KAU colleges. TiMELY, which is based on text mining techniques, analyzes and compares course descriptions when uploaded as Word, Text, Pages, JPEG, GIF, and JPG file formats. Then, the tool works in automated and manual mode to support a higher level of accuracy. It performs data extraction based on predefined keyword relevancy and allows a person in charge of the process, called the responsible equivalency, for the course equivalency process to modify the keywords if needed. TiMELY autonomously decides whether to equalize the given course for the transferred student.

With all the mentioned issues to illustrate the need for an automated system in the previous section, the main challenge for TiMELY is to support text mining and recognition techniques efficiently for the Arabic language. This work implements a DSS to develop a new tool called (TiMELY) using Text-Mining based on course EquivaLencY procedures. DSS assists in analyzing massive amounts of data to enhance decision makers' judgments during the equivalency process [6]. We incorporated Arabic in addition to English languages into TiMELY because the Arabic language is widely used, and it is the official language in Saudi Arabia and 25 other countries in the region. A total of 300 million people speak Arabic, and the number will increase exponentially. Several other languages also use the Arabic alphabet, such as Urdu, Hebrew, Persian, Jawi, and Kurdish [7]. The rapid technological evolution in AI has increased technological companies' interest in developing a practical tool for Arabic text mining and recognition [7].

The Arabic alphabet differs in the way it is drawn, and the reading direction from right to left makes the text mining techniques for other languages unsuitable for the Arabic language. Arabic language text recognition is complicated because of the cursive and overlapping writing nature of the characters (Fig. 1)

and many dots and diacritics that differentiate between letters [7]. Diacritics (Ta-ShØ-ki-lØ: تشكيل) are short vowels known as (Dammah, Kasra, Fatha, Sukun, Shaddah, Tanwin, Hamza, Maddah, Alif Khanjariyah) and written on top or underneath a character that could change the meaning of a word (we elaborated more with examples on Tab. 1) (Fig. 1) [4–8]. Furthermore, the Arabic language has 28 characters, each of which has two to four shapes according to its position in a sentence. For example, the letter (Yaa: ي) has four different representations: word initially indicating a verb in the present tense, (Ya-dØ-ru-sØ: يدرس: he is studying); word medially representing a noun to indicate dual, (Da-rØ-sai-nØ: درسين: two lessons); word finally specifying ownership of a noun (Da-rØ-si: درسي: my lesson); and in an isolated form (Mi-hØ- wa-ri: محوري: pivotal). Furthermore, unlike most languages, the Arabic language is written from right to left. Moreover, if one of the characters (DaaØl: د), (Alif: ا), (ZaaØl: ذ), (Raa: ر), (zaaØy: ز), and (Waa/w: و) occurs mid-word, then the word becomes separated into two or more sub-words. The possibility of overlapping also increases, especially with a specific type of fonts; for example, “Diwani” and “Andalusi” (Fig. 1 and Tab. 1) [9]. The large number of available Arabic fonts complicates the text recognition process as well.

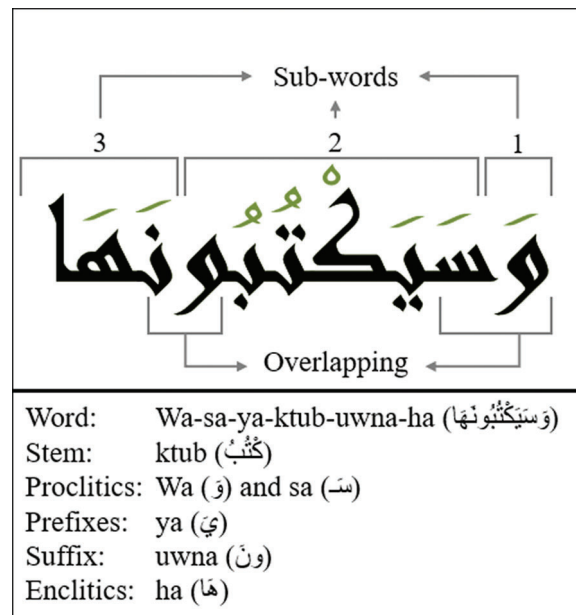


Figure 1: Example of semantic difficulties for Arabic word structure

Table 1: Examples of Arabic diacritic complexities

#	Word with diacritics in Arabic	Transcription	Diacritic position and type with the letter La:m	Example
1	أَلِف	A-li-fØ	Kasra underneath the letter La:m	The letter Alif is the first letter in the alphabet
2	أَلْف	A-lØ-fØ	Fatha on top the letter La:m	Ten decades is equal to a thousand-year
3	أَلَف	A-lØ-la-fØ	Shadda on top of the letter La:m	The author composed a new book

More complexities exist at the morphological level. For example, the single word (Wa-sa-ya-ktub-uwna-haa: وَسَيَكْتُبُونَهَا: then, they will write it) [10] consists of different morphemes described in Fig. 1 that we redesigned, which emphasizes details to highlight the complexity. The general structure of an Arabic word is as follows: [Proclitic(s)+[Prefix(es)]] + stem + [Suffix(es)+[Enclitic]]. The stem is the heart of all words, such as in Fig. 1 example (KØtub: كَتَبَ), while affixes are the prefix and suffix that can be attached to the stem, such as (Yaa: يَ) and (Uwna: وَنَ). By contrast, the clitics are one to three letters that symbolize the proclitic and enclitic, such as (Waa: وَ), (Saa: سَ), and (Haa: هَا) in the example [11].

In the segmentation and extraction method, the Arabic language poses several difficulties during the automation and processing, which requires the involvement of a human actor to edit the result of a processed document manually. Moreover, building a bilingual database requires accurate translation and mining capabilities to achieve acceptable efficiency.

## 2 Related Work

Many prestigious universities, such as McGill University, have attempted to solve course equivalency problems using their automated systems. An educational DSS typically explores data using analytical models to generate decision makers' options to improve policies, strategies, planning, and monitoring of the educational system. The most dominant tools are explained below, and we also summarized the advantages and disadvantages of Tab. 2. The work used limited data from two Saudi Universities; with big data, different techniques would be recommended to use unlimited data; another limitation is related to the generated information from the comparison considering that no accredited official body exists for the reported results.

### 2.1 King Abdul-Aziz University Decision Support Center (KAU DSC)

The KAU DSC is located in Saudi Arabia, and its objective is to assist the quality of scientific research and the educational process. It mainly provides information about the University's programs' standards and regulations and performance indicators for decision makers. It also improves the value of students' learning and the University's utilities' quality. The data at KAU DSC were collected from all the University's sectors and modeled to support its strategic objectives. The KAU DSC course equivalency system is still under development.

### 2.2 MHE's Certificate Equalization System

The Ministry of Higher Education of Kuwait uses an online equivalency system to equalize certificates and makes it available to every University. The user must have a registered account to log in and benefit from the system's services even before uploading a certificate to the system, which is stored and equalized manually. Finally, the equalization results are saved in the server database.

### 2.3 McGill University's Course Equivalency System

McGill University's Course Equivalency System is available in Canada to support extracting course information within the University or across educational institutions. The system has a database that contains the course names and descriptions for various universities around the world. The system provides users with clear instructions on using and storing the University's database's equalization results. Unfortunately, the equalization process is performed manually, and the system only supports the English language.

As shown in Tab. 2, current available DSSs, as discussed above, do not complete the equalization process for courses electronically and require more human involvement in the loop. It is contrary to our proposed tool, TiMELY, which can automate the entire course equalization process and limit human intervention to assist in training the algorithm to only the process' final phases.

**Table 2:** Comparison among educational tools

# Systems	Advantages	Disadvantages
1 King Abdul-Aziz University Decision Support Center (KAU DSC)	<ul style="list-style-type: none"> <li>• Support Arabic language</li> <li>• Helps in decision making</li> <li>• Enhance the quality of the University's utilities</li> <li>• Place the collected data into use to strategic objectives</li> </ul>	<ul style="list-style-type: none"> <li>• Only limited for KAU use</li> <li>• The system is not fully automated</li> <li>• Does not serve the course equivalency process</li> </ul>
2 MHE's Certificate Equalization System	<ul style="list-style-type: none"> <li>• Specialized in the course equivalency process</li> <li>• Support Arabic language</li> </ul>	<ul style="list-style-type: none"> <li>• Only limited to Kuwait's University</li> <li>• The equivalency process is conducted manually</li> <li>• Time consuming</li> </ul>
3 McGill University's Course Equivalency System	<ul style="list-style-type: none"> <li>• Specialized in the course equivalency process</li> <li>• Support Canadian and American universities</li> </ul>	<ul style="list-style-type: none"> <li>• Does not support the Arabic language</li> <li>• The equivalency process is conducted manually</li> <li>• Time consuming</li> </ul>
4 TiMELY	<ul style="list-style-type: none"> <li>• Specialized in the course equivalency process</li> <li>• Support Arabic and English languages</li> <li>• Uses AI techniques (text mining and recognition)</li> <li>• Automated system</li> <li>• Help students in decision making</li> <li>• Minimize the errors in the equivalency process</li> </ul>	<ul style="list-style-type: none"> <li>• Needs academic affair approval (a human intervention)</li> <li>• Needs more data training in Arabic and English</li> </ul>

### 3 Proposed Method

On the basis of the interviews, survey, and observation, the course equivalency system is conducted manually. It involves mainly three agents: the applicant, academic affairs, and responsible equivalency. First, we designed a survey targeting the 80 transfer students in all faculties during the first semester to determine their need for an automated system and the end-user interaction. Then, we interviewed the five academic affair representatives, plus the equivalency responsible in all faculties, and the deanship of distance learning and e-learning based on their defined tasks to fully understand the existing system and perceive their expectations.

We found that 72% of students were unsure of the number of credit hours gained after equalization and the remaining credit hours in the study plan when transferring to another college or major. A total of 90% of students thought that an automated system for course equivalency would facilitate the procedure if it is applied before the transfer request. From an academic affair viewpoint, the process and procedure are time and effort consuming. We went further and decided to investigate the number of available decision support tools for the Arabic language and found only two (mentioned in Section 4).

We started to build a dataset repository of 1000 keywords for selecting the optimal algorithm to bridge the existing gap. Thereafter, we collected 50-course descriptions chosen randomly from the Faculty of Computing and Information Technology, Faculty of Economics and Administration, Faculty of Law, and Faculty of Arts and Humanities at KAU and Faculty of Finance at King Faisal University. The datasets were divided equally between Arabic and English course descriptions. We tested the developed dataset on keyword extraction algorithms, and the results showed that the keyword extraction using the word co-occurrence algorithm [11] scored the highest in F1 score [12], with 90% for English course descriptions and 80% for Arabic course descriptions. The results showed that the accuracy rate was insufficiently good, and the word extraction process should be improved and increased more efficiently. TiMELY uses the following user interface tools:

### 3.1 User Interfaces

The design of TiMELY interfaces follows the “10 heuristics” for user interface design [13]. TiMELY is also a centralized decision support tool that operates among all the end-users: students, academic affairs, and the responsible equivalency. It learns and accelerates the updating and analysis of data. We show the interactions between the TiMELY tool and all the end-users in Fig. 2.

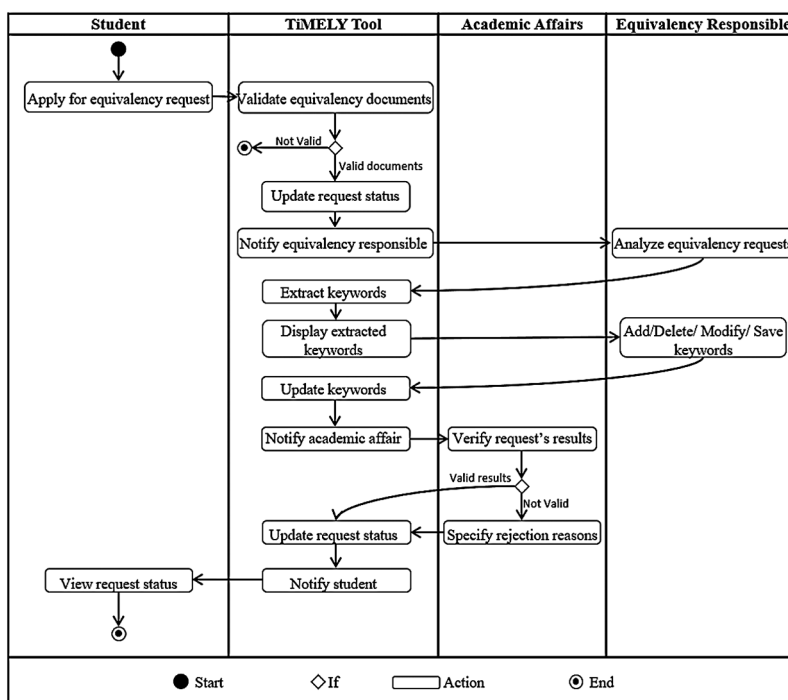


Figure 2: TiMELY workflow

The process starts when a student applies for a new equivalency request in our tool. TiMELY will validate the uploaded documents that request to check if the uploaded course description matches existing courses' descriptions. In case the course description does not exist or the student uploaded false documents, then the tool will reject and terminate the request; otherwise, it will proceed. Next, TiMELY will update the request status and notify the responsible equivalency for analyzing the courses' contents.

Thereafter, TiMELY will display the extracted keyword list to the equivalency responsible. Then, the equivalence responsible will choose to add, delete, modify, and save the keyword list. TiMELY will

automatically update the keyword list, calculate the percentage of matched keywords, and upload the course description and the transferred department's course. Subsequently, the tool will notify the academic affairs to verify if the resulting percentage is valid according to the University's regulations. If the request did not fulfill all of the required University's rules and regulations, then the academic affairs would specify the rejection reasons, or he or she will accept the request, which will proceed and trigger TiMELY to update the status and notify the student about the completion of the process.

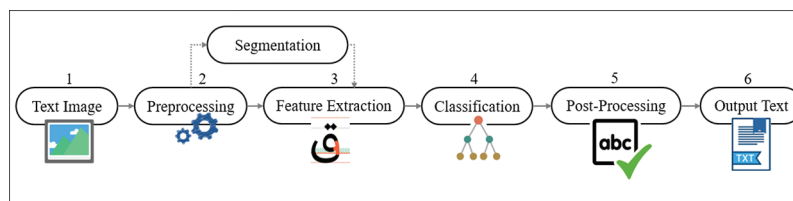
## 4 Results

### 4.1 Technical Feasibility

The main aim of technical feasibility is to complete the project from a technical perspective [6,14,15]. The following section describes and compares different techniques of text mining and recognition.

### 4.2 Text Recognition

The text recognition process consists of five steps, as illustrated in Fig. 3. The first step is pre-processing; it applies specific algorithms to the input picture or data to reduce noise for simplifying the next step [16]. The second step is text segmentation that defines the task of dividing the text into segments, where a page is segmented into letters. Each segment is topically coherent, and cutoff points indicate a topic change [17].



**Figure 3:** Text recognition steps

A third step is an analytical approach called feature extraction. It is a critical aspect of Arabic text recognition to achieve a high accuracy performance. It also distinguishes attributes of the segmented characters for use in the fourth step, which is the classification step [9]. Such technique identifies and assigns the segmented features into predefined classes. Each class provides a set of elements with some type of similarity, where it could be beneficial for storing keywords and essential information in a database [7].

The fifth step is post-processing, which enhances the recognition efficiency by correcting linguistic spellings and parsing the unstructured text [9]. Parsing is the process of breaking up ordinary unstructured text and simplifying it. The unstructured text refers to a plain text file with no predefined format in different forms, such as XML or HTML [12].

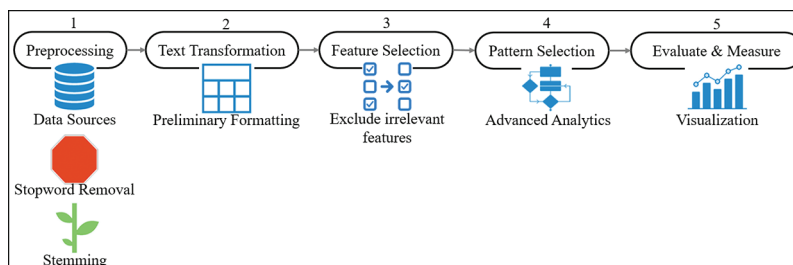
### 4.3 Text Mining

The text mining process saves time and effort and helps in supporting decisions [4]. We demonstrate the workflow of the five phases of text mining implementation in Fig. 4.

First is the pre-processing phase or the post-processing phase in text recognition. It begins by filtering all text of a given document, removing redundancies, eliminating words such as “a, an, the” plus conjunctions such as “and, but,” and stemming the words. Stemming is a natural grouping of words with similar meanings—the process of reducing the Affix to Suffix, Prefix, or root [4–18]. Second, the text transformation phase defragments the text to a small unit called features (identifying features of the documents). This phase will generate a massive number of features preparing for the next step. The third phase is called feature selection;



it aims to minimize database space and simplify the searching technique by eliminating irrelevant features. Those features could affect the algorithms' logic and accuracy [19]. Fourth, the pattern selection phase stores extracted data from the previous phase in a structured database. The final phase is to evaluate and measure the results of the process [18–27].



**Figure 4:** Text mining steps

#### 4.4 Algorithm Analysis

We collected a dataset with 1000 keywords (500 Arabic and 500 English) during our data gathering process. Then, we manually specified the list of keywords. Thereafter, we tested three keyword extraction algorithms: term frequency-inverse document frequency (TFIDF), Cortical.io tool with Retina Database, and word co-occurrence algorithm. We examined and compared the results using validation technique calculations, namely, precision (P), recall (R), and F1 score. Then, we computed four measurements, as shown in Tab. 3, which we designed based on [12–28].

**Table 3:** Measurements and equations

Name	Definitions	Equation
True Positive (TP)	Number of resulting keywords that match the correct keywords	
False Positive (FP)	Number of resulting keywords that do not match the correct keywords	
True Negative (TN)	Number of negative keywords that do not generate result after applying the algorithm	
False Negative (FN)	Number of correct keywords that do not generate result after applying the algorithm	
Precision (P)	The percentage of the true result against performance of incorrect information retrieval	$P = TP / TP + FP$
Recall (R)	The percentage of the true result against performance of forgotten information retrieval	$R = TP / TP + FN$
F1 score	A measure that combines precision and recall	$F1 = 2 (P * R / P + R)$

The value returned for precision (P) represents the accuracy for the extracted keywords by the algorithm, where the recall (R) represents the level of relevancy, and the value of the F1 score combines precision and recall. Our aim while testing the algorithm performance was to obtain higher values for precision (P) and recall (R) (high F1 score).



#### 4.5 Experimental Results

From multiple existing algorithms and tools used in text mining, we selected the most suitable algorithm to build our tool, namely, TiMELY.

1. Term frequency-inverse document frequency (TFIDF) measure algorithm [12] is the most famous statistical algorithm for extracting keywords that appear frequently. It works directly on the top of the fetched document and treats it as the “*corpus*.” TFIDF is robust and efficient for the dynamic tenor because any change in the document requires updating prorated frequency counts. It is typically efficient for a massive quantity of text documents, such as in scoring and ranking documents in search engines.
2. Cortical.io tool with Retina Database [20] is an application of semantic folding, which is a new model for natural language understanding that simulates how the human brain processes the text to numerical representation named the semantic fingerprint, which allows intuitive text comparisons. The Retina Database contains actual terms distributed over a  $128 \times 128$  grid matrix as a 2D structure. The topic location on the matrix in the semantic space mimics the neocortex.
3. Keyword extraction using word co-occurrence algorithm [11] after the text pre-processing phase and stop word removal. The algorithm first generates a co-occurrence matrix to count the frequency for each pair of terms. For example, if a term occurs in 11 different sentences, then the value of the terms of co-occurrence is equal to 11. Accordingly, the algorithm creates a co-occurrence matrix by  $\chi^2$  (chi-square measure) value; its equation is  $(N \times G)$ , where N is the number of unique terms in the text and G is the frequent terms denoted as the set of the frequent. Thereafter, the algorithm measures linguistic relations between words and extracts keywords. The co-occurrence algorithm focuses on short contexts and quantifies the degree of bias during the co-occurrence circulation [21]. Frequent terms would have a significant meaning if the term’s bias degree scored a high  $\chi^2$  value.

According to the experimental results in Tabs. 4 and 5, the keyword extraction using word co-occurrence algorithm achieved the highest performance for Arabic and English databases. TFIDF did not score well for course description documents because of the shortness of text; most of the course descriptions were around 500 words or less. Furthermore, the Cortical.io tool with Retina Database achieved the lowest score, particularly for the Arabic language.

**Table 4:** Test results for recall and precision for English descriptions

Algorithms	TP	FP	FN	Precision	Recall	F1M
Term Frequency-Inverse Document Frequency (TFIDF)	506	57	73	0.90	0.87	0.88
Cortical.io tool with Retina Database	483	112	97	0.81	0.83	0.82
Keyword Extraction Using Word Co-occurrence	517	39	62	0.92	0.89	0.90

**Table 5:** Test results for recall and precision for Arabic descriptions

Algorithms	TP	FP	FN	Precision	Recall	F1M
Term Frequency-Inverse Document Frequency (TFIDF)	278	127	108	0.68	0.72	0.70
Cortical.io tool with Retina Database	191	209	195	0.48	0.50	0.49
Keyword Extraction Using Word Co-occurrence	314	86	72	0.79	0.81	0.80

Word co-occurrence accuracy in the Arabic text needs improvement. We enhanced the algorithm's results by developing an Arabic stop word list and involved a human intervention during the course equivalency process. The equivalency responsible will approve the extracted keywords or modify them to train the algorithm before TiMELY saves the results. The TiMELY results are shown in [Tab. 6](#).

**Table 6:** TiMELY test results for recall and precision

Algorithms	TP	FP	FN	Precision	Recall	F1M
Keyword Extraction Using Word Co-occurrence (English)	179	2	2	0.98	0.98	0.98
Keyword Extraction Using Word Co-occurrence (Arabic)	152	2	4	0.97	0.99	0.98

#### 4.6 Testing Techniques

Testing is the procedure of debugging and checking errors when executing a program. It validates and verifies whether the system has met all the requirements [22].

##### 4.6.1 Unit and Integration Testing

Unit testing consists of several encapsulated classes, and every class contains different methods. We performed unit testing using the primary methods [23]. [Tab. 7](#) presents one of the methods' results of the extraction function in our tool TiMELY. We examined and validated the data using the xUnit open-source framework. It considers the latest unit testing technology in C# [24].

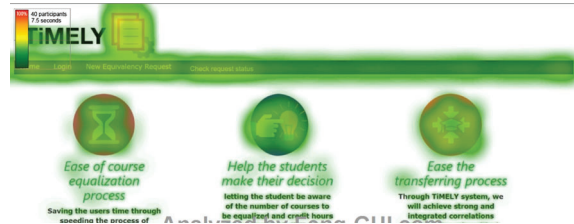
**Table 7:** Example of unit tests for the extracting keyword function

# Test case description	Expected result	Actual result	Pass/Fail
1 Test case 1 checks the method of text extraction from a text file format the student's uploads	Return true values	Return true values	Pass
2 Test case 2 checks the method of text extraction from a pdf file format the student's uploads	Return false values	Return false value	Pass
3 Test case 3 checks the extraction classes if the method extracts the keywords from the text	Return true values	Return true values	Pass
4 Test case 4 checks the modification if the method stores the responsible equivalency keywords that he or she adds	Return true values	Return true values	Pass

##### 4.6.2 Usability Testing

We tested the interfaces using the Feng-GUI tool, which is smart testing in measuring visual efficiency and eligibility [25–29], to ensure the usability of TiMELY. It works by mimicking a 5 seconds eye-tracking session for 40 end-users and develops advanced reports that predict human eye movement similar to the heat map report [23–25]. This click tracking heat map report is a useful tool for collecting data about users' clicks. It visualizes areas that receive most of the user's attention and interaction. The heat map colors depend on the design and type of the website. If TiMELY used advertisement or were an e-commerce website, then the emphasis will be focused (more red areas) on the advertisement picture. TiMELY's interface design aims to spread the user's focus on all the page contents given that it is a DSS tool, which results in more green

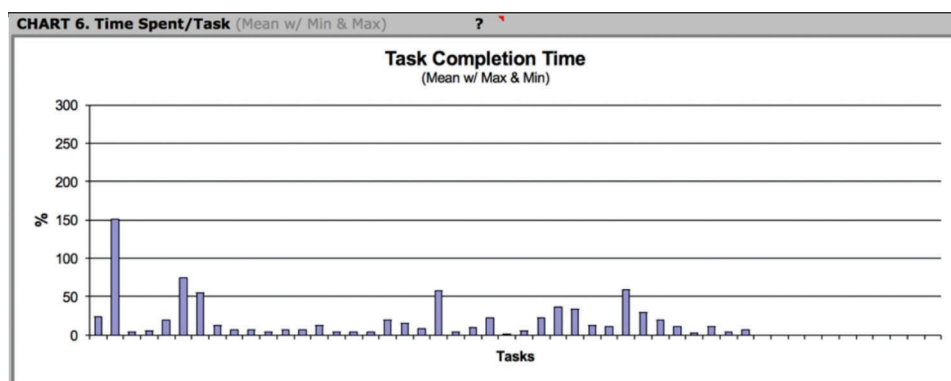
areas. We tested TiMELY's home interface, as shown in Fig. 5. The test showed that the user's attention was equally divided on the page, which validates the usability of the interface's design.



**Figure 5:** Result of heat map test for the TiMELY home interface

#### 4.6.3 Learnability and Performance

We implemented a usability test for measuring learning and performance by using 20 users from academic affairs, responsible equivalence, and students. We first performed a usability pilot study that gives us an overall knowledge of the user experience with the first version of TiMELY to improve it. Second, all the users were asked to create a new request, login, and check the request; for each request, we measured and recorded the time per user. Finally, the users answered the System Usability Scale (SUS) questionnaire for measuring the system's usability and user satisfaction in 10 sentences to evaluate their experience with TiMELY (Fig. 6). The users evaluated their experience with TiMELY from 1 (do not agree) to 5 (agree), and the success score was calculated depending on their responses. The questionnaire's result was a 92.5% success rate, which indicates the usability in TiMELY and that users find the tool useful, learnable, and easy to use.



**Figure 6:** Performance & time response results

## 5 Conclusion and Future Work

We briefly presented an overview of the course equivalency process and its problems and discussed the Arabic language challenges. We illustrated our developed bilingual decision support tool called TiMELY for course equivalency using AI techniques (text mining and recognition). We described all algorithms and functions tested using unit and integration testing. We also applied usability tests such as the SUS questionnaire and heat map test for the end-users. The developed TiMELY tool scored 92.5% for overall test performance. We recommend continuous development to overcome the system's current limitations using advanced AI techniques including web mining, deep learning, and knowledge representation for unlimited data given that these technologies are always evolving. Furthermore, our tool is currently

bilingual to extend the system's implementation to cover various applications serving establishments, companies, and government agencies that use Arabic text documentation. Our tool also needs more enhancement to score higher **F1M**. We suggest adding multi-languages and test the tool using the Google platform or IBM Watson and creating an innovative unified segmentation process linked to phonological transcription in Arabic and similar languages.

**Funding Statement:** This work is funded by the Deanship of Scientific research (DSR), King Abdulaziz University, Jeddah, under grant No. (DF-618-165-1441). The authors, therefore, gratefully acknowledge DSR technical and financial support.

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

- [1] N. Ifill, A. W. Radford, J. Wu, E. F. Cataldi, D. Wilson *et al.*, "Persistence and attainment of 2011–12 first-time postsecondary students after 3 years. first look." Washington, DC: National Center for Education Statistics, 2016.
- [2] R. K. Strader, "On education, formation, citizenship, and the lost purpose of learning," *International Journal of Christianity & Education*, vol. 23, no. 2, pp. 241–242, 2019.
- [3] King Abdulaziz University "Electronic admission portal- transfer to the university," [Accessed: 02-Jun-2020]. [Online]. Available: <https://transfer.kau.edu.sa/>.
- [4] M. A. Mohammed, S. S. Gunasekaran, S. A. Mostafa, A. Mustafa and M. K. Abd Ghani, "Implementing an agent-based multi-natural language anti-spam model," in *International Symposium on Agent, Multi-Agent Systems and Robotics (ISAMSR)*, IEEE, Putrajaya, pp. 1–5, 2018.
- [5] S. A. Mostafa, A. Mustapha, A. A. Hazeem, S. H. Khaleefah and M. A. Mohammed, "An agent-based inference engine for efficient and reliable automated car failure diagnosis assistance," *IEEE Access*, vol. 6, pp. 8322–8331, 2018.
- [6] M. A. Mohammed, M. K. Abd Ghani, S. A. Mostafa and D. A. Ibrahim, "Using scatter search algorithm in implementing examination timetabling problem," *Journal of Engineering and Applied Sciences*, vol. 12, pp. 4792–4800, 2017.
- [7] A. M. Al-Shatnawi, F. H. Al-Zawaideh, S. Al-Salameh and K. Omar, "Offline Arabic text recognition - an overview," *World of Computer Science and Information Technology Journal (WCSIT)*, vol. 1, no. 5, pp. 184–192, 2011.
- [8] A. Al-Zayyat, "al-Mu'jam al-Wasit," *Majma Al-Lughat Al-Arabiyya*, 5th ed. Dar Kotob al-Ilmiyya & Maktaba al-Shuruq al-Duwaliyya, 2004.
- [9] S. A. Mostafa, M. S. Ahmad, A. Mustapha and M. A. Mohammed, "Formulating layered adjustable autonomy for unmanned aerial vehicles," *International Journal of Intelligent Computing and Cybernetics*, vol. 10, no. 2, pp. 11–08 2017.
- [10] S. N. Srihari and G. Ball, "Guide to ocr for Arabic scripts," *An Assessment of Arabic Handwriting Recognition Technology*, Verlag, London: Springer, 2012.
- [11] Y. Matsuo and M. Ishizuka, "Keyword extraction from a single document using word co-occurrence statistical information," *International Journal on Artificial Intelligence Tools*, vol. 13, no. 01, pp. 157–169, 2011.
- [12] H. H. A. Razaq, A. S. Gaser, M. A. Mohammed, E. T. Yassen, S. A. Mostafad *et al.*, "Designing and implementing an Arabic programming language for teaching pupils," *Journal of Southwest Jiaotong University*, vol. 54, no. 3, pp. 0258–2724, 2019.
- [13] M. Beaudouin-Lafon, "Human-computer interaction, beyond human-computer interaction," *New Paradig*, vol. 1, pp. 227–254, 2006.
- [14] Salunke, D. Badhe, V. Doke and Y. Raykar, "The state of the art in text recognition techniques," *Ijarccce*, vol. 4, no. 2, pp. 442–444, 2015.
- [15] L. Wolfe, "How to write a professional technical feasibility study," [Accessed: 01-May-2020]. [Online]. Available: <https://www.thebalancecareers.com/writing-technical-feasibility-study-3515778>.

- [16] S. Das, R. Kumar, R. Cherku and S. Gutta, "Segmentation of overlapping text lines, characters in printed telugu text document images," *International Journal of Engineering, science and technology*, vol. 2, no. 4, pp. 6606–6610, 2010.
- [17] O. Koshorek, A. Cohen, N. Mor, M. Rotman and J. Berant, "Text segmentation as a supervised learning task," *Association for Computational Linguistics, Louisiana. 2000*, vol. 1, pp. 469–473, 2018.
- [18] R. Balamurugan and S. Pushpa, "A review on various text mining techniques and algorithms," 2nd Int. Conf. on recent innovations in science, engineering, and management JNU convention center, New Delhi: Jawaharlal Nehru University, pp. 838–844, 2015.
- [19] Oracle Database Online Documentation Library "Data mining concepts, text mining," [Accessed: 04-Jun-2020]. [Online]. Available: [https://docs.oracle.com/cd/E11882\\_01/datamine.112/e16808/text.htm#DMCON409](https://docs.oracle.com/cd/E11882_01/datamine.112/e16808/text.htm#DMCON409).
- [20] E. S. Francisco, "Semantic folding: theory and its application in semantic fingerprinting," in *Semantic Folding Theory*, 2nd ed., Vienna: Cortical.io, *arXiv preprint arXiv: 1511.08855*, Mar 2016.
- [21] M. Al-Kabi, H. All-Belaili, B. Abul-Huda and A. Wahbeh, "Keyword extraction based on word co-occurrence statistical information for Arabic text," *Abhath Al-Yarmou K: Basic Sci. & Eng*, vol. 22, no. 1, pp. 75–95, 2013.
- [22] M. Williams, "Microsoft visual c#(tm). net (core reference)," Microsoft Press, Washington DC, 2002.
- [23] R. S. Pressman, "Software engineering: A practitioner's approach," 7th ed., vol. 9781118592. New York: McGraw-Hill Companies, Inc, 2014.
- [24] NET Foundation "About xUnit, xUnit.net," [Accessed: 04-Jun-2020]. [Online]. Available: <https://xunit.net/>.
- [25] Feng-GUI "Ai-powered design audit tool for digital marketing agents," [Accessed: 22-Mar-2020]. [Online]. Available: <https://feng-gui.com>.
- [26] S. A. Mostafa, S. S. Gunasekaran, A. Mustapha, M. A. Mohammed and W. M. Abdullallah, "Modelling an adjustable autonomous multi-agent internet of things system for elderly smart home," in *Proc. AHFE*, Washington, pp. 301–311, 2019.
- [27] O. A. Mahdi, Y. R. B. Al-Mayouf, A. B. Ghazi, A. W. A. Wahab and M. Y. I. B. Idris, "An energy-aware and load-balancing routing scheme for wireless sensor networks," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 12, no. 3, pp. 1312–1319, 2018.
- [28] K. H. Abdulkareem, M. A. Mohammed, S. S. Gunasekaran, M. N. Al-Mhiqani, A. A. Mutlag *et al.*, "A review of Fog computing and machine learning: Concepts, applications, challenges, and open issues," *IEEE Access*, vol. 7, pp. 153123–153140, 2019.
- [29] A. Mutlag, M. Abd Ghani, N. Arunkumar, M. Mohammed and O. Mohd, "Enabling technologies for fog computing in healthcare IoT systems," *Future Generation Computer Systems*, vol. 90, pp. 62–78, 2019.