An Automatic Arabic Essay Grading System based on Text Similarity Algorithms

Abdulaziz Shehab, Mahmoud Faroun, Magdi Rashad
Information Systems Dept.
Faculty of Computers and Information, Mansoura University, Egypt

Abstract—Manual grading process has many problems such as time consuming, costly, enormous resources, lot of efforts and huge pressure on instructors. These problems place the educational community in a dire need to have auto-grading systems so as to address these problems associated with manual grading. Auto-grading systems are wide spread over the world because they play a critical role in educational technology. Additionally, the auto-grading system is introducing many advantages as it saves cost, effort and time in comparison to manual grading. This research compares the different algorithms used in automatic grading systems in Arabic languages using string and corpus algorithms separately. This process is a challenging task following the necessity of inclusive assessment to authenticate the answers precisely. Moreover, the challenge is heightened when working with the Arabic language characterized by complexities in morphology, semantics and syntax. The research applied multiple similarity measures and introduces Arabic data set that contains 210 students’ answers. The results obtained prove that an automatic grading system could provide the teacher with an effective solution for article grading systems.

Keywords—Auto-grading systems; string-based similarity; corpus-based similarity; N-Gram

I. INTRODUCTION

Nowadays, auto-grading systems are a very important scientific topic within the educational community. The huge amounts of tests and students have brought about the necessity for automatic grading systems. Therefore, researchers pay more attention to such systems which have become a vital component in the educational community because it is capable of reducing the load of the teaching staff due to graded students’ exams. These systems have witnessed a continuous development in the last few years [1]-[5]. In fact, most of the teachers’ time is wasted due to manual grading, teaching staff all over the world suffer from wasted time spent on students’ essay marking.

There is an increase in the number of student enrolment that makes the marking operation more difficult. Essay Questions are more difficult than other objective questions because it takes much time to mark these questions. But, the essay questions are preferable than objective questions because it makes the cheating process more difficult and raises the student’s skills in writing.

In addition, the essay marking can differ from human grader and others in marks, which is unfair. Therefore, the auto-grading systems could automatically grade the student answers and teachers could utilize the time wasted for marking the essays. In this way, teachers get an opportunity to concentrate on other critical assignments, for example, develop more viable instructional materials and other correspondence capacities.

Essay-type questions are classified into two main types: short and long answers. For the first, student solutions are written in short sentences. For the latter, it gives the students the freedom to write as long as he or she wants. Hence, teachers look for special characteristics to be graded, for example, style of writing, mechanics, and content [6]. The short answers assessment depends on the content of answers and the style is not significant [7].

Auto-grading system has one of the most convoluted tasks as it depends on the overall semantic meaning, since short answers have many common words [8]. Moreover, the similarity on long essay is complicated to discover as every word could have other synonyms, meanings [9], [10]. The marks could be given to the students based on similar sentences that are prepared in the marking scheme. The majority of automated short answer grading systems are done in English ignoring Arabic because there are many challenges in Arabic those needs to be tackled in different ways. Arabic context has unique characteristics and features so it is challenging task.

In the last few years, many researchers have proposed a number of automated Arabic short answer grading systems [11]-[14]. They proposed many algorithms like Latent Semantic Analysis (LSA) [15], Damera-levenshtein (DL) [16], [17], N-Gram [18], Extracting Distributionally similar words using Co-occurrences (DISCO) [19], [20]. The main idea of automatic grading systems is using n-grams in different applications [21]-[23]. More recently, there are approaches that used LSA model to evaluate written answers automatically [21], [22], [24]-[26]. However, their proposals lack a comparative study that deeply shows the advantages and disadvantages of such algorithms.

Dependently, in this paper, a comparative study between different approaches that are oriented for automatic grading is presented. The main goal of this paper is to find the best suitable technique suitable for Arabic language essay questions. So, we applied four algorithms on the same dataset of questions to compare them and find the optimal algorithm which gives the best correlation between manual grading and automatic grading. The remainder of this paper is as follows: Section II discusses related works; Section III presents the
used dataset. Section IV presents a comprehensive view on string-based and corpus-based while Section V presents the proposed architecture. The experimental results are shown in Sections VI and VII is our conclusion and future work.

II. RELATED WORKS

There are several automatic grading systems conducted in English such as E-rater [27], Automark [28], C-rater [29], and LVQ [30]. On the other hand, a few studies have been done in Arabic [4], [11], [12].

Gomaa and Fahmy [11], one of the most recently published works, present a system that scores Arabic short answers. They prepared a dataset which contains 610 student answers written in the Arabic language. Their proposal evaluates the student answers after they have been translated into English. Their objective was to overcome the challenges that were faced in Arabic text processing. However, the proposed system has many problems namely the missing of utilization of good stemming techniques, the translation from Arabic into English causes the loss of context structure where many words in Arabic are not semantically translated, and the results obtained have to be passed to a machine learning that demands a high processing time.

Mezher and Omar [12] proposed a hybrid method based on a modified LSA and syntactic feature in a trial for automatic Arabic essay scoring. They relied on the dataset proposed in [11]. Their proposal focused on part of speech (POS) tagging in a way to identify the syntactic feature of words within the similarity matrix. Their study sought to resolve the drawbacks of standard LSA which laid emphasis on the limited syntactic analysis. However, utilizing only LSA technique in their study did not guarantee a high correlation ratio.

Emad [13] presents a system based on stemming techniques and Levenshtein edit operations for evaluating student’s online exams. The proposed system was mainly based on the capabilities of light and heavy stemming. The dependence only on the string-based algorithm (Levenshtein) is counted as one of the main defects in his study as it ignores corpus-based algorithms that support semantic similarity.

Rashad et al. [31] proposed an Arabic online examination environment that provides an easy interaction between students and instructors. In fact, their proposal is only developed for grading objective (non-essay) questions. For essay grading, the proposed is nothing more than a storage system where the instructor has to manually assess the student’s writings.

Alghamdi et al. [14] present a hybrid automatic system that combines LSA and three linguistic features: 1) word stemming, 2) word frequency, and 3) number of spelling mistakes. Their proposal determine the optimal reduced dimensionality used in LSA to evaluate the performance of their proposed system. Khalid and Izzat [4] present a method depending on synonym and stemming of words. They assign weights the instructor’s answer words to benefit the assessing process. Their study was impracticable and had neither a dataset nor an experimental result.

III. DATASET

In this research, general methodology will be used to advance a grading system of Arabic short answers based on text similarity matching methods. To evaluate the methods for short answer grading, we used the dataset prepared in a general sociology course taken by secondary level 3 students where the total short answers in this dataset are 210 short answers (21 questions/assignment x 10 student answers/question). It was evaluated by judges where the scores ranged between 0 (completely wrong) and 5 (completely right). Each judge was unaware of the other’s correction and grade. We considered the average score of two annotators as the gold standard to test the automatic grading task. Table I displays a sample question, model answer, student answers, and the average score.

<table>
<thead>
<tr>
<th>Question</th>
<th>Model Answer</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>عفر USAGE (Arabic)</td>
<td>عفر USAGE (Arabic)</td>
<td>5</td>
</tr>
<tr>
<td>عفر USAGE (Arabic)</td>
<td>عفر USAGE (Arabic)</td>
<td>2</td>
</tr>
<tr>
<td>عفر USAGE (Arabic)</td>
<td>عفر USAGE (Arabic)</td>
<td>3</td>
</tr>
<tr>
<td>عفر USAGE (Arabic)</td>
<td>عفر USAGE (Arabic)</td>
<td>2.5</td>
</tr>
<tr>
<td>عفر USAGE (Arabic)</td>
<td>عفر USAGE (Arabic)</td>
<td>1.5</td>
</tr>
<tr>
<td>عفر USAGE (Arabic)</td>
<td>عفر USAGE (Arabic)</td>
<td>1</td>
</tr>
<tr>
<td>عفر USAGE (Arabic)</td>
<td>عفر USAGE (Arabic)</td>
<td>5</td>
</tr>
<tr>
<td>عفر USAGE (Arabic)</td>
<td>عفر USAGE (Arabic)</td>
<td>5</td>
</tr>
<tr>
<td>عفر USAGE (Arabic)</td>
<td>عفر USAGE (Arabic)</td>
<td>0.5</td>
</tr>
<tr>
<td>عفر USAGE (Arabic)</td>
<td>عفر USAGE (Arabic)</td>
<td>2</td>
</tr>
</tbody>
</table>

IV. STRING-BASED AND CORPUS-BASED APPROACH

A. String-Based Text Similarity

Damerau-Levenshtein (DL) algorithm works on counting the quorum of processes that are required to map one string into another string. These operations could insert, or a character obliterate from the string. Moreover, it could be a replacement of a single character, or a reversal of two adjacent characters. This method is based on counting the number of replacements and insertions required to transform one string into another string.

The Damerau-Levenshtein distance between two strings is defined as the minimum number of operations required to transform one string into another string. The operations are:

1. Insertion: Insert a character into the second string.
2. Deletion: Delete a character from the first string.
3. Substitution: Replace a character in the first string with a different character in the second string.
4. Transposition: Swap two characters in the first string.

The Damerau-Levenshtein distance between two strings is the minimum cost of any sequence of these operations that transforms one string into the other.

The Damerau-Levenshtein distance is calculated as follows:

\[ D(L, R) = \min_{S} \left\{ C(S) : S = O_1 \rightarrow O_2 \right\} \]

where \( D(L, R) \) is the Damerau-Levenshtein distance between strings \( L \) and \( R \), \( S \) is a sequence of operations, and \( C(S) \) is the cost of the sequence of operations.

The cost of insertion, deletion, substitution, and transposition is calculated as follows:

1. Insertion: \( C(\text{Insert}) = 1 \)
2. Deletion: \( C(\text{Delete}) = 1 \)
3. Substitution: \( C(\text{Substitution}) = 1 \)
4. Transposition: \( C(\text{Transposition}) = 2 \)

The Damerau-Levenshtein distance is calculated as follows:

\[ D(L, R) = \min_{S} \left\{ C(S) : S = O_1 \rightarrow O_2 \right\} \]

where \( D(L, R) \) is the Damerau-Levenshtein distance between strings \( L \) and \( R \), \( S \) is a sequence of operations, and \( C(S) \) is the cost of the sequence of operations.
characters [16]. In fact, DL is not only limited to these four operations but also it could treat 80% of all human misspellings [16, 17] (Hall & Dowling, 1980; Peterson, 1980). To compute the DL similarity value (DLSim), the DL distance is normalized through the following equation:

\[
DL = \frac{MaxLength - DLDis}{MaxLength}
\]

where MaxLength is the extreme length of the 2 strings and DLDis is the obtained DL space between these two strings.

N-gram algorithm [32] works on sliding a window of length n over string in order to generate a number of 'n' length grams that are utilized in the matching process. Dependentely, a match is then compared to N-gram matches within the other string. Hence, when 2 strings s1 and s2 are within a small edit space of each other, they will share a large number of N-grams in common. For example, the positional q-grams of length q=3 for string "مصر
cốn trong
" are f(1,مصر,
), f(2,مصر,
), f(3,مصر,
), (مصر). Prefixes indicate the beginning of the words: antefixes, prefixes, suffixes and postfixes. One can categorize them according to their syntactic role. Antefixes are generally prepositions agglutinated to the conjugation person of verbs in the present tense. Arabic word stemming is a technique that finds the lexical root or stem for words in natural language, by removing affixes attached to its root, because an Arabic word can have a more complicated form with those affixes. An Arabic word can be represented after stemming process as ذهب.

Several types of affixes are agglutinated at the beginning and the end of the words: antefixes, prefixes, suffixes and postfixes. One can categorize them according to their syntactic role. Antefixes are generally prepositions agglutinated to words at the beginning.

Suffixes are the conjugation ending of verbs and the dual/plural/female markers for the nouns.

Finally, postfixes represent pronouns attached at the end of the words. All these affixes should be treated correctly during word stemming [33].

V. PROPOSED SYSTEM

The proposed system is based on measuring the similarity of student answer by comparing each word in the model answer with each word in student’s answer using a bag of words (BOW) model to produce the final automatic score. Several string and corpus algorithms run individual answers to obtain similarity values. Fig. 1 shows the steps of the systems.

A. Raw

The similarity in Raw method is computed without applying any Natural Language Processing (NLP) task. Stemming techniques are applied to Arabic words to extract the triliteral roots of words.

B. Tokenization

The first step in the pre-processing is Tokenization, where it divides the text sequence into sentences and the sentences into tokens. In alphabetic language, words are usually surrounded by whitespace. Besides the whitespace and the commas, the tokenization also removes {({[]):; })} from the text and presents the words in the model answer. Tokenizing is the process of separating each word in the document which becomes the basis of the word, removing prefix, insertion, suffix and duplication.

C. Stopwords

As a pre-processing step for all of the fourteen string similarity measures, removes ineffective common words. Stop-words filters out common words that do not have significant meaning to measuring the similarity. In our system, the stop words are removed according to a predefined list that has 378 words in Arabic. This process is aimed to get the word to represent the content of the document.

These are typically what linguists would call function words, consisting mostly of a relatively small class of articles ("،", ",", etc.), prepositions ("على", "في", etc.), pronouns ("هو", ",", etc.). These stop words does not convey a meaning.

D. Stemming

Arabic word stemming is a technique that finds the lexical root or stem for words in natural language, by removing affixes attached to its root, because an Arabic word can have a more complicated form with those affixes. An Arabic word can be represented after stemming process as ذهب.

Several types of affixes are agglutinated at the beginning and the end of the words: antefixes, prefixes, suffixes and postfixes. One can categorize them according to their syntactic role. Antefixes are generally prepositions agglutinated to words at the beginning.

Prefixes, usually represented by only one letter, indicate the conjugation person of verbs in the present tense.

Suffixes are the conjugation ending of verbs and the dual/plural/female markers for the nouns.

Finally, postfixes represent pronouns attached at the end of the words. All these affixes should be treated correctly during word stemming [33].

\[
\text{DlDis} = \min _{n} \{\text{MaxLength} - \text{MaxLength}'(s1) - \text{MaxLength}'(s2)\}
\]
The objective of stemming is to find the representative indexing form of a word by truncating these affixes.

Antefixes (ٗتاه, ٗاه, تاه, فاه, ماه, ٗىو, اه, ٗب, ٗه, ىو, فس, ٗفة, ٗف, ك, ف, ٗب, ه)

Prepositions meaning respectively: and with the, and the, then the, as the, to (for), then will, then with, then to (for), and will, as, then, and, with, to (for).

Prefixes (ا, ُ, ٙ, خ)

Letters inflecting morphologically for person.

Suffixes (تَا, يُ٘, تيِ, تاُ, اخ, اُ, ُٗ, يِ, ٗا, تٌ, تِ, ّا, خ, ُ, ا, ٙ, ٗ)

Word endings inflecting for verb ending, and dual/plural/female markers

Postfixes (كما, هما, كن, هن, تى, ها, نا, هم, كم, كن, م)

Pronouns meaning respectively: your, their, your, their, my, her, our, their, your, your, his, me.

E. Stopstem

A combination of stop-words and stemming tasks are applied.

VI. EXPERIMENTAL RESULTS

Our experiments were performed using two string-based algorithms and two corpus-based algorithms. Four different methods Stop, Raw, Stop-Stem and Stem are used in testing for string-based algorithms. However, only the Stop method is used in corpus-based algorithms where Raw, Stem, and Stop-Stem methods cannot be utilized because there is no need to measure the semantic similarity between the Stop words.

The correlation constant was calculated between automatic system and the human grading. The correlation coefficient is used to determine to what extent the system and the human are correlated in assigning the grades.

Equation (1) is correlation constant where X, Y are two sets and \(\bar{x}, \bar{y}\) are the averages of each set in series.

\[
Cor(X,Y) = \frac{\sum (x-\bar{x})(y-\bar{y})}{\sqrt{\sum (x-\bar{x})^2 \sum (y-\bar{y})^2}}
\]  

As shown in Fig. 2 and 3 the correlation between the applied algorithms and the manual grading are presented for the same question. The main target of any algorithm is to be very near to the manual grading to prove the efficiency of this technique. The degrees are from 0 to 5 and the number of students is ten. The figures show that the N-gram can be used as the nearest algorithm to the manual grading.

As noticed from Table II, in string-based distance measures; DL resemblance got the best association value 0.803 practical to the Stop-Stem text. The reason behind this could be Stop stem works on the origin of the words comparing the characters against the model answer and neglects the stop-words. This enables the algorithm provide very high results.
Moreover, for N-gram algorithm, the best correlation achieved while using Stop method is 0.820. The character-based N-gram algorithm achieved better results than the other three types. In general, the character-based N-gram approach has many advantages, such as: simplicity; it is more reliable for noisy data such as misspellings and grammatical errors; it outputs more N-grams in given strings than N-grams resulting from a word-based approach, which leads to collecting a sufficient number of N-grams that are significant for measuring the similarity. The paper proved that using string algorithms gives the teacher an effective solution to help them to undertake student grading with high precision. The paper presented the comparison between four effective algorithms to prove the possibility of automatic grading over manual grading systems.

In future work, we are aiming to combine the string algorithm and corpus algorithm together to achieve the highest possible results depending on the synonymous method and content method to decrease automation grading errors. We are aiming to use different datasets in different subjects to find out the reliability of those algorithms in applications.

**REFERENCES**


**TABLE II. THE CORRELATION RESULTS BETWEEN MODEL ANSWER AND STUDENT ANSWERS USING DL, N-GRAM, LSA, AND DISCO2.**

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Raw</th>
<th>Stop</th>
<th>Stem</th>
<th>Stop-Stem</th>
</tr>
</thead>
<tbody>
<tr>
<td>DL</td>
<td>0.799</td>
<td>0.800</td>
<td>0.798</td>
<td>0.803</td>
</tr>
<tr>
<td>N-Gram</td>
<td>0.800</td>
<td>0.820</td>
<td>0.796</td>
<td>0.795</td>
</tr>
<tr>
<td>LSA</td>
<td>N/A</td>
<td>0.781</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Disco2</td>
<td>N/A</td>
<td>0.796</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>


