Journal of King Saud University – Computer and Information Sciences xxx (xxxx) xxx

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

Towards the automatic generation of Arabic Lexical Recognition Tests using orthographic and phonological similarity maps

Saeed Salah [a,*], Mohammad Nassar [a], Raid Zaghal [a], Osama Hamed [b]

[a] Department of Computer Science, Al-Quds University, Jerusalem, P.O. Box 20002, Palestine
[b] Computer Systems Engineering Department, Palestine Technical University, Tulkarm, Palestine

ARTICLE INFO

Article history:
Received 12 September 2020
Revised 30 January 2021
Accepted 5 February 2021
Available online xxxx

Keywords:
NLP
LRT
N-gram
Dialects
MSA
Orthographic
Phonological

ABSTRACT

Lexical Recognition Test (LRT) themes are one of the main methods that are widely used to measure language proficiency of some common languages such as English, German and Spanish. However, similar research for Arabic is still at development stages, and existing proposals mainly use human-crafted methods. In this paper, a new methodology, based on a newly developed algorithm, was proposed with the aim of automatically constructing high quality nonwords associated with a real quick measurement of Arabic proficiency levels (Arabic LRT). The suggested algorithm will automatically generate nonwords based on Arabic special characteristics they are orthography (spelling), phonology (pronunciation), n-grams and the word frequency map, which is an important factor to create a multi-level test. With the help of a large dataset of Arabic vocabulary, the proposed algorithm was experimented. For this purpose, a Web-based application, following the suggested methodology, was designed and implemented to facilitate the process of collecting and analyzing learners' responses. The experimental results have shown that the LRT questions that were automatically generated by the proposed system had confused the learners, this is clear from the output of the confusion matrix which showed that (1/3) of the generated nonwords were able to distract the learners (with accuracy 65%). Consequentially, the results of recall and precision have smaller values, 0.52 and 0.48, respectively.

1. Introduction

Arabic is one of the main spoken languages being widely used nowadays. It is among the top seven languages around the world. More than 422 million people consider this language their native language, and many others use Arabic for other purposes such as understanding religion, collecting peoples' opinions, and studying Arab cultures, etc. (Abdelgadir and Ramana, 2017; Farghaly and Shaalan, 2009). Arabic is divided into three main categories, *classical*, *spoken*, and *standard* (Elfardy and Diab, 2012; Habash, 2017). Classical Arabic is the legacy language that had been used in the ancient time; spoken Arabic "the dialect" represents a high diversity of the real spoken language among Arab regions (Levant, Moroccan, Egyptian and Gulf) leading to so-called an Arabic "Diglossia", which means that people use the same word to express different purposes; standard Arabic is the official language being used in language learning centers and books. Both classical and spoken Arabic have their own specificities and usage and are rarely used for scientific research. Therefore, standard Arabic is the core part of this research work, and most of the Arabic language research had been implemented using Modern Standard Arabic (MSA) corpora that was derived from formal news agencies, books, social media, and religious books.

Several natural language processing proposals emphasized the need to have efficient mechanisms to measure Arabic language proficiency through simple and fast placement testing methods (Hamed and Zesch, 2018a; Hamed, 2019). Therefore, the enthusiasm to enrich Arabic language research using Natural Language Processing (NLP) technologies is highly demanded by the research community on these days.

NLP is a set of techniques that interact between Artificial Intelligence (AI) and linguistics (Nadkarni et al., 2011). These techniques can be used for several purposes such as machine

* Corresponding author.
E-mail addresses: sasalah@staff.alquds.edu (S. Salah), mohammmad.nassar@gmail.com (M. Nassar), zaghal@staff.alquds.edu (R. Zaghal), osama.hamed@ptuk.edu.ps (O. Hamed).

Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

https://doi.org/10.1016/j.jksuci.2021.02.006
1319-1578/

Please cite this article as: S. Salah, M. Nassar, R. Zaghal et al., Towards the automatic generation of Arabic Lexical Recognition Tests using orthographic and phonological similarity maps, Journal of King Saud University – Computer and Information Sciences, https://doi.org/10.1016/j.jksuci.2021.02.006

translation, text mining and processing, spelling auto corrections, Optical Character Recognition (OCR) applications, sentimental analysis, generation of automated languages placement tests, test autocorrections, voice recognition and others (Menacer et al., 2017; Abdul-Mageed, 2017). There are many research efforts that tackled Arabic NLP (ANLP), both written and spoken parts (Farghaly and Shaalan, 2009; Habash, 2010; Guellil et al., 2019). Some research efforts like (Jarrar et al., 2017; Bougrine et al., 2017; Al-Twairesh et al., 2018) proposed solutions to handle Arabic dialects by collecting each vocabulary from these dialects into one corpus. Others like Salloum and Habash, (2014) and Hegazi, (2016) tackled Arabic diacritical marks; and the orthography of Arabic script, which is used to indicate the style of writing Arabic letters with to letters' positions (first, middle, and last). Many letters have different shapes when located in various positions. Dialects and diacritical marks are beyond the scope of this research work.

As described by Hamed and Zesch (2017), Lexical Recognition Test (LRT) is a vocabulary size test that is frequently used for measuring language proficiency worldwide. In such a test, the learners are being shown either valid words like "denial" or nonwords like "platery", and they need to decide if they are valid or invalid words. The main advantage of the LRT is its simplicity; it only takes five minutes to answer all questions. As shown in Fig. 1, only "Yes/No" or checklist questions are asked, and the scores can easily be automated. The current problem could be noticed when learners cannot find an effective measure to indicate their levels of recognition and knowledge in Arabic language. Therefore, this research contribution aims at developing Arabic LRT based on a newly proposed algorithm that automatically generates main skeleton of the LRT, which is already applied to other European languages. The proposed algorithm will follow certain rules to generate nonwords with high quality, *i.e.,* a nonword that could confuse the learner, and add a level of complexity to the LRT. Also, this method applies generic machine learning paradigms, namely character n-gram models to generate good nonwords for Arabic LRTs. We also applied other methodologies that could fit better with Arabic special characteristics related to Arabic phonetic and orthographic. Generally, phonetic declares how characters could be spoken and morphology indicates how characters are being written. Furthermore, we used the original word frequency map, which is an important factor to generate various complexity levels of the LRT.

Also, the research problem could be addressed while we are going to design an Arabic LRT by extending the current algorithms being extensively used in other languages like English, German, Spanish, etc. This situation induced several factors to be involved and considered as special properties of Arabic language (a right to left written language) (Hamed and Zesch, 2015, 2017). Moreover, Arabic has multi-millions of concreate used and unused vocabularies – about 12 Million, this restricts the abilities to create good nonwords, which have high ranking similar to concreate words to be included in the Arabic LRT. Besides the above challenges, research in ANLP is still at early stages of development, *e.g.,* the lack or scarcity of resources (corpora). There are a few commercial corpora that can be used to help in conducting a relevant research. Considering more than one corpus implies that more available words exist, this will reduce error rate when classifying the generated vocabularies to be either nonwords or concreate words.

Fig. 1. Examples of questions used in LRT in both English and German languages.

Therefore, this research contribution aims at developing Arabic LRTs in an automatic way by proposing a new algorithm that considers these problems. It is supported by implementing a Web-based application that generates such placement tests to measure learners' proficiency levels. The main hypothesis that this research work is assumed to achieve are generating Arabic nonwords based on similarities, both writing and pronunciation, will distract the learner, especially when considering original word frequency map and n-grams concepts. Also, including new nonwords in a special corpus, to be used in generating of Arabic LRT will enhance the test's results and performance. Therefore, the main contributions of this work are:

- Proposing a new algorithm that considers some Arabic language characteristics to automatically generate high quality nonwords that increase the complexity/difficulty of Arabic LRTs.
- Developing a Web-based application based on the proposed algorithm to ease the management and collection of learners' responses and analysis.
- Developing a validation criterion to evaluate the accuracy of the proposed approach. The validation was mainly based on human-intervention, a version of the test was written by Arab experts following the same rules, and the obtained results were compared, analyzed, and discussed.

Besides the introduction section, this paper contains the following sections. A review of the related work is presented in Section 2. Section 3 details the proposed methodology and the suggested algorithm. The used dataset, the evaluation measures, and the experimental results and their discussions are presented in Section 4. Finally, in Section 5, we conclude this research work and provide some future works.

2. Related work

Linguists had been the focus of several research efforts with the aim of finding the best way that could help language learners to know their proficiency levels. For example, English Lexical project (Balota et al., 2007) contains international standard tests that had been created and became a standard measurement criterion to assess learner' proficiency levels for a specific language. Among these tests are Test of English as a Foreign Language (TOEFEL) and The International English Language Testing System (ILETS). These two tests are widely used to measure English language proficiency levels for various categories of academic and business classifications. Another short quick test that had been used to give an indication towards English learner proficiency and other Latin languages is the LRT. Many experiments and research, coming from different research centers in Europe, had worked on this type of research to prove this concept using a real test implementation. Since this research work focused on Arabic LRT, in the following we discuss the most relevant contributions, and shed light towards their main drawbacks. Consequently, we avoided the potential problems related to some similar experiments (Khalil and Darwish, 1967; Balota et al., 2007) carried out to design this test previously. Thus, this historical background associated with each entity related to creating Arabic LRT as the components that have an effect while generating good nonwords like diacritization role (Jarrar et al., 2017; Hamed and Zesch, 2017) and its benefits.

LexTALE is a measurement for language proficiency applied for English, Dutch and German languages (Balota et al., 2007; Lemhöfer and Broersma, 2012). LexTALE is a five minute (YES, NO) vocabulary identification test; it shows good results when

indicating a vocabulary dataset, but it is still substantial when comparing it with other language proficiency tests like TOEIC (Test of English for International Communication), where users could apply this test through accessing this Website (https:// www.lextale.com). LexTALE test consists of 60 (YES, No) questions, 40 words and 20 nonwords. Nonwords were generated and created manually, but the process of generating these nonwords should be efficient, and the generated nonwords must look like real words that could distract foreign learners from identifying them easily. LexTALE is considered as a good measurement for nonnative English language speakers having levels from medium to high. LexTALE for Dutch and German are still not classified as a good measure. The manual generation of LexTALE tests is also available. This manual process creates nonwords by replacing certain characters within the target word to obtain a similar nonword in terms of orthographic and phonological concepts. Validation of the generated LexTALE tests was done by correlating its' results with other proficiency measurement tests such as Quick Placement Test (QPT). The test has been adapted to other languages beyond English, *e.g.,* Dutch, and German, French, and Spanish (Duyck et al., 2004).

A manual generation of nonwords was adopted by English Lexicon Project5 (ELP) (Balota et al., 2007). ELP is a large repository of databases (descriptive and behavior), it is linked to a search engine that aims to supply researchers with the necessary resources that could help them overcoming the faced obstacles of processing the lexical tests. ELP could be accessed through the following Web site (https://elexicon.wustl.edu). Data were collected from 1300 participants from six universities. Some of the exploratory information about this dataset is shown on the Website of the project that was mentioned above, it provides additional descriptive statistical data for the available words and nonwords and their frequencies. The ELP uses a manual procedure to create nonwords through replacing certain characters within the target word to obtain a nonword which is similar to the original one in terms of orthographic, phonological, and morphological. In the ARC nonword database (Rastle et al., 2002), the researchers provided a model based on phonological and orthographical rules that were applied to English of southern British. The results of this application are presented on the Website of this project with some statistical information. Items in this database were used to build the LRT test that is intended to strike the learner in different ways based on the morphological, orthographical, and phonological rules.

Wuggy research project (Keuleers and Brysbaert, 2010) proposed a computer application that help researchers creating a better quality pseudoword or nonword following rules of languages, sub syllabic structure and transition frequencies between sub syllabic elements were used. It is already applied for multiple languages like Dutch, English, German, French, Spanish, Serbian, and Basque, and it could be expanded to other languages with some extra efforts. In this regard, pseudoword is considered as an important factor for lexical decision that represents a major tool used by psycholinguists to perform word processing tasks. Some of the limitations of the Wuggy algorithm are *(i)* it mainly depends on sub syllabic or summed bi-gram similarities; *(ii)* the program requires a user input called matching expression, so it is not fully automated solution for nonwords generation; *(iii)* the algorithm does not auto-detect the expression by which the word is ending.

Another similar application to Wuggy, called WordGen, was implemented. It is a tool for nonword selection and generation used in Dutch, English, German, and French (Duyck et al., 2004). In this research both manual and automatic methods were used to generate nonwords. Other researchers (Jarrar et al., 2017; Hegazi, 2016; Hamed and Zesch, 2015) tried to show the important role of Arabic diacritized towards vocabulary assessment in the LRT, as they believed that diacritization reveals words ambiguity and makes better judgement while learners identify the words. For this purpose, a sample for diacritized version of Arabic lexical test was generated along with a non-diacritized version to show the role of diacritization. The results have shown that the absence of diacritization increases the ambiguity of word's identification. It is worth mentioning that the most written text in Arabic is a non-discretized, except in some historical, religious, and classical books, as well as in some specialized Arabic educational domains. Diacritization has an impact on nonwords design as Arabic diacritization is an orthographical way to describe Arabic word pronunciation (Khalil and Darwish, 1967). We hypothesize that the non-diacritized nonwords are probably more difficult than

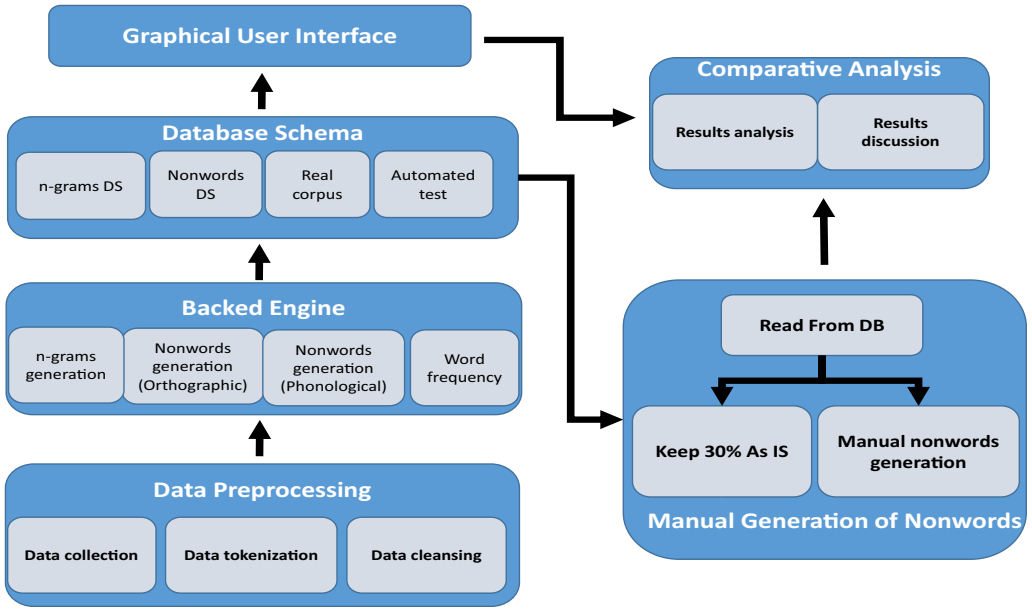

Fig. 2. The proposed block diagram of Arabic Lexical Recognition Test (LRT).

the diacritized ones. The diacritized nonwords can distract learners better with closely related words, especially if they are labelled with pronounceable diacritics.

Hamed and Zesch (2015) suggested the use of a fully automatic methodology to generate high quality nonwords that could be used as stimuli in English LRTs and confuse the learners. To apply the fully automated process of generating nonwords in English language, the authors conducted a study of good nonwords generation using automatic methods by replacing a letter through an algorithm based on Markov and character language models, besides that the authors had ranked the generated nonwords and used the highest ones in English LRT. Another similar work was carried out by Rastle et al., (2002). The authors implemented an automatic nonword generation paradigm for English language by building a database of nonwords based on both phonetical and orthographical language properties.

Gueddah and Yousfi (2013) proposed an approach to improve the process of spell checking and correction when typing documents in Arabic. The proposed approach suggested the use of statistical model to find Arabic letters' similarity degree using a similarity matrix in which each letter has a weighted degree of similarity with other Arabic letters by associating costs to the permutation errors involving the proximity of keyboard characters and the calligraphic similarity in Arabic alphabet. Their main aim was to design a spell checker for erroneous words committed in typing of documents in Arabic language. Compared to this research work, we found a specific similarity matrix for each letter based on the Arabic phonological and orthographic characteristics, so reputations will be performed based in a small set of similarities. Furthermore, the two works have different scopes, the main goal is to have an Arabic LRT. While they main aim was to propose an Arabic spellchecker.

Compared to the previous research efforts, this research work differs in the way that the approach being proposed considers the process of generating the nonwords in automatic way based on algorithmic approach considering four Arabic language characteristics: Orthography – spelling, phonology – pronunciation, n-grams, and the word frequency map, which is an important factor to create a multi-level test. To generate nonwords, we have been inspired by its definition: "words that fulfill the phonological constraints of the language but do not bear meaning" (Huibregtse et al., 2002). One of the approaches used to generate nonwords in English is minimal pairs (Ricks, 2015), a corresponding way to do it in Arabic is the use of orthography and phonology. It has been noted by Hamed and Zesch (2015) that frequent n-grams are highly likely to generate good nonwords, which look-like existing words. It has been also noted by Ellis (2002) that the frequent words are easier to guess than less frequent words. In addition to that nonwords in Arabic could be described as fake Arabic vocabulary that looks like a real word, and it was designed to distract the learner and confuse him/her in terms of phonetical if he/she tried pronunciation, and orthographical in terms of word writing shape. These arguments were derived from Al-Ain book of the author al-Khalil ben Ahmad al-Farahidy (Khalil and Darwish, 1967).

3. The proposed approach

Fig. 2 shows the main steps of the suggested methodology. In the following, we discuss the backend engine that handles the following two main sub-sections:

3.1. Nonwords generation: orthographic and phonologic

The automatic process of generating the nonwords is based on generating nonwords with the help of the above Arabic language characteristics, *i.e.,* orthographic, phonological, n-grams, and vocabulary frequency. Algorithm 1 shows the procedure for generating the nonwords. The proposed algorithm begins by looping through all cleaned vocabularies stored in the database. For each vocabulary, it calculates its frequency. To generate multilevel tests, the algorithm calculates the word frequency (Frequency); how many times the selected word appeared in the corpus. Two thresholds were used ($Threshold_1$ and $Thresholds_2$) to tune the algorithm's operation. If Frequency > $Threshold_1$ && Frequency < $Threshold_2$ – the vocabulary is not used more frequently, the algorithm generates two lists: L_p; the list of orthographical vocabularies based on orthographical similarity map, and L_o; the list of phonological vocabularies based on phonological similarity map (Refer to Table 1). Next, it adds the two lists together to form a similarity list (SimilarityList), which contains all vocabularies generated from both orthographical and phonological similarity maps. To generate test's questions, the algorithm randomly selects vocabulary from the SimilarityList, and checks the occurrence of this vocabulary in the processed dataset (ProcsDSList). If the conditional statements return FALSE – this means that the selected vocabulary is a nonword, it adds it to the NonwordList to be used by the LTR test. If the condition statement returns TRUE – this means that the selected vocabulary is considered as a real word, it removes it from the SimilarityList, and repeats the process again by selecting a new random vocabulary from the SimilarityList. For each generated nonword, the data record will store the ID of the original one, the replacement letter, the replacement position, and the new letter.

3.2. Generating n-grams

To improve the process of generating nonwords, the results of Algorithm 1 have been refined by applying character n-grams concepts, which are the subsequent characters of vocabulary. This function loops through the cleaned data file, and then for each vocabulary, it generates all possible n-grams starting from bi-gram to word-length-1 g. These n-grams were inserted into a database table with respect to the real word, this might be helpful to formulate a statis-

Algorithm 1: The proposed algorithm for nonwords generation

```
start procedure
1. Initialize: NonwordList()=null, ProcDSList,
   SimilarityList= null, Frequency, Threshold1,
   Threshold2
2. // First step: Read random word from
   ProcDSList
3.   loop // For each word in ProcDSList
4.     word = getNewWord()
5.       Frequency = ProcDSList.count(word)
6.         if (Threshold1 < Frequency < Thresh-
   old2) {
7.       LO = ListofOrthographics(word)
8.       LP = ListofPhonologics(word)
9.       SimilarityList= LP+LO
10. endif
11. Nonword =getRandomWord(SimilarityList)
12. if (ProcDSList.find(Nonword) == False)
13.   NonwordList.add(Nonword)
14. else
15.     SimilarityList.del(Nonword)
16.     goto step (11)end procedure
```

Table 1
The orthographical and phonological similarity maps of Arabic letters.

Similarity type	Similarity set
Orthographic	ح،ج،خ
Orthographic	ب،ت،ث
Orthographic	س،ص،ش
Orthographic	ذ،د
Orthographic	ض،ذ،ظ،ض
Orthographic	ق،ك،ف
Orthographic	ع،غ
Phonological-Place of articulation (velar-الحلق)	غ،ع،ح،خ، ه،ء
Phonological-Place of articulation (glottis-اللسان)	ت،ث،ج،د،ذ،ر،ز،س،ش،ص،ض،ط،ظ،ق،ك،ل،ث،ي
Phonological-Place of articulation (bilabial-الجوف)	ب،ف،م،و
Phonological-Place of articulation (oral cavity-الشفتان)	أ،و،ي

tical data reference throw which we can build some conclusions and judgements. Since n-grams could be involved in generating nonwords by replacing a character in the input word taking into consideration frequency occurrence of prefix and postfix characters. Thus, the nearest character from the similarity group intersected with a letter that uses frequency in the n-grams list will be substituted. This way, n-grams are being used to narrow the acceptable possibilities; this is expected to increase the quality of the nonword generation process.

The following is an example taken from the new n-gram dataset that is generated when the word "حاليا" – which means presently – is being fed as an input to the algorithm.

['حا : ha', 'ال : al', 'لي : ly', 'يا: ya', 'حال : hal', 'الي: aly','ليا: lay', 'حالي: holy', 'اليا: alyan', 'حاليا: halyan']

All words are collected in one file and are persisted in relational database tables, while dealing with structured database is generally easier and faster. This oracle database schema was used in building and manipulating the relevant LRT test, analyzing the results, and building the needed reports and dashboards, etc. For each word in the database table, we built a query with the needed Oracle SQL aggregative functions to retrieve word frequency to be inserted into a new table that holds word and its frequency (distinct values). As mentioned above, word frequency had been considered when selecting test items; thus, the highest frequent words are the most common words, consequently, they will be easy to guess. From the opposite side, nonwords of high frequent words will be easy to guess, and they will not confuse the learner. Therefore, the test items will have frequency less than the average to be robust enough.

To ease understanding the proposed approach (Fig. 2), we provide an illustrative example to see all the steps in action. By referring to Fig. 2, the first step is to have a raw dataset. We picked up the following sentence taken from KACST Corpus (Table 2, Reference [3])

هنا سواء أينعت أو أثمرت"، وأضاف "سأكتب سيرتي ذات يوم وأتحدث عن الأصدقاء وحتى الأعداء الذين لهم مكان في قلبي".

The next step is data tokenization which was applied to separate the content of each data file using a whitespace as a delimiter. The output of this step is the following list of words:

هنا | سواء | أينعت | أو | أثمرت"، | وأضاف | "سأكتب | سيرتي | ذات | يوم | وأتحدث | عن | الأصدقاء | وحتى | الأعداء | الذين | لهم | مكان | في | قلبي."

Next, it comes the data cleaning step. It is the process of eliminating any undesired text content including punctuation marks, special symbols, Arabic dialects and diacritical marks, numeric values, stop words, one char length items, and any strange items. The output of this step is a list of clean words.

هنا | سواء | أينعت | أو | | أثمرت | وأضاف | سأكتب | سيرتي | ذات | يوم | وأتحدث | عن | الأصدقاء | وحتى | الأعداء | الذين | لهم | مكان | في | قلبي

The list of pronunciations of the above list of words is

Hona/here | Sawa'/wether | Eyna't/grow up | Aw/or | Athmarat/get floured | Wa adaf/ and add | Sa'ktob/I will write | Serti/my life story | That/in a | Yawm/day | Wa-atahath/and talk | An/about | Al-asdika'/the friends | Wa hata/and even | Al-ada'/the enimes | Al-ladina/who | Lahoum/have | Makan/place| Fe/in | Kalbi/my heart.

After data preprocessing steps, it comes the backend engine steps where all n-grams, phonological and orthographic similarity lists were extracted for each word. For simplicity, we proceed with the example considering one sample word قلبي / kalbi/my heart, where it appears 30 times in the whole corpus (Word frequency = 30). An example of phonological similarity is كلبي (replacing kkaf: Q:ق with kaf:k:ك), an example of orthographical similarity is فلتي (replacing kkaf:Q:ق with fa':F:ف), and the list of the n-grams is [قل/all | لب/lub | بي/be | قلب/qalb/heart | لبي/lbe | قلبي/qalbi/my heart].

All extracted similarity lists (phonological, orthographic and n-grams) and frequency of the word قلبي are fed as inputs to the next stage where the database scheme is built. Here, the n-gram database stores n-grams for each word in the database table with respect to the original word. Nonword database stores all possible nonwords (phonological, orthographic) with respect to the original word, and the real corpus stores all distinctive real words with respect to their frequency values.

4. Experimental results and discussions

4.1. Dataset and data preparation

In this research study, freely available corpora datasets had been used. They were collected from different resources, such as news agencies, social media, and Arabic books; these files were used in similar projects that have been tackled the manual generation of Arabic LRT (Hamed and Zesch, 2017; Rastle et al., 2002). Taking into considerations that there are other paid resources and as per our research purpose this free source is adequate to implement such experiment, and it is considered by other relevant research (Hamed, 2019). In the following, we summarize some of the exploratory information about the used dataset. The dataset contains a huge number of Arabic texts in raw format. The collected files were transformed to one UTF-8 format having one word per line. Some preprocessing operations were applied to convert the data into suitable format to work with. We mainly applied data cleaning to eliminate special symbols, non-Arabic characters, numeric values, punctuations, whitespaces, and any other strange character. Table 2 shows some technical information about the used dataset. The first column of this table represents the corpus source; the free source from which the data was obtained, each source could have one or more files, number of characters, lines as in a notepad^{++} text file, size in (KB), diacritized or not diacritized, and the main reference.

Referring to Table 2, it can be clearly seen that data files belonging to "Watan source" have occupied the most shares, and the total number of diacritized and non-diacritized words is 16.3 Million and 18.5 Million, respectively. We observed that the collected dataset has some redundancy. We argue that data redundancy will have a significant value when generating nonwords based on the original word frequency map, which is an important factor to determine the test level. Since frequency has an inverse correlation with difficulty of the generated nonwords, and this conforms to the argument that the most common the word is, the easier to be known, and it is not easy to confuse the learner when replacing a letter with its similarity.

Fig. 3 shows the various preprocessing steps that were applied on the raw dataset. Since the gathered data are in raw format, the

Table 2
Summary of the raw dataset (dimensions and references).

Corpus source	File name	Char count	Lines	Size [KB]	Diacritized
Al-Jazeera Corpus[1]	aljazeera.txt	13,260,976	80,369	13,058	No
Al-Jazeera Corpus[1]	aljazeera100.txt	977,321	5,887	955	No
Books Corpus[2]	books.txt	858,622	1,533	839	No
KACST Corpus[3]	KACST.TXT	24,551,235	74,106	23,976	No
KACST Corpus[3]	KACST100.txt	1,077,781	74,106	1,053	No
Al-Khaleej-2004 Corpus[4]	khaleej.txt	27,283,987	5,695	26,645	No
Al-Khaleej-2004 Corpus[4]	Khaleej100.txt	1,106,419	231	1,081	No
Al-Watan-2004 Corpus[4]	Wata100.txt	1,043,107	178	1,019	No
Al-Watan-2004 Corpus[4]	Watan.txt	124,202,282	178	121,292	No
Watan Diac Corpus[4]	Watan-diac.txt	163,473,924	40,579	159,643	Yes
Quran[5]	quran.txt	743,918	6,236	727	No
RDI[6]	rdi.txt	858,844	2,579	839	No
Tweets[7]	Tweets-ann.txt	1,528,273	10,007	1,493	No
Tweets[7]	Tweets-sharp.txt	1,514,713	10,007	1,480	No
WikiNews[8]	WikiNewsTruth.txt	177,279	423	174	No
Total		**362,658,681**	**312,114**	**354,274**	

1 URL: http://www.aljazeera.net/portal [Online; Last Accessed 29th, July 2020].
2 URL: https://sourceforge.net/projects/tashkeela/ [Online; Last Accessed 29th, July 2020].
3 URL: https://sourceforge.net/projects/kacst-acptool/files/ [Online; Last Accessed 29th, July 2020].
4 URL: https://sites.google.com/site/mouradabbas9/corpora [Online; Last Accessed 29th, July 2020].
5 URL: http://tanzil.net/download/ [Online; Last Accessed 29th, July 2020].
6 URL: http://www.rdi-eg.com/RDI/TrainingData/ [Online; Last Accessed 29th, July 2020].
7 URL: https://www.aclweb.org/anthology/D15-1299 [Online; Last Accessed 29th, July 2020].
8 URL: https://www.aclweb.org/anthology/W17-1302 [Online; Last Accessed 29th, July 2020].

tokenization process was applied first to tokenize the content of each data file using a whitespace as a delimiter. This process is necessary to have each word in a separate line, and then to accumulate all results into one text file. During the tokenization process, the procedure eliminated punctuation marks, special symbols, Arabic diacritization, numeric values, stop words, one char length items, and any strange items. All cleaned words from all data sources had been stored into one text file, where each word is stored in a single line.

A data refinement operation was also applied on the dataset to eliminate diacritization. As illustrated in Table 3, the collected diacritized dataset is less than the non-diacritized part. Therefore, eliminating diacritization will enhance the non-diacritized dataset, and contribute to having enough amount of data to be considered as a final corpus data. All target data were collected in one text file with UTF-8 encoding, some statistical details are displayed in Table 3 which shows the total number of words extracted from each file. Table 4 shows that the average word length is (6.5), and hence, the generated test has a query condition that determines this range of length to formulate the test items.

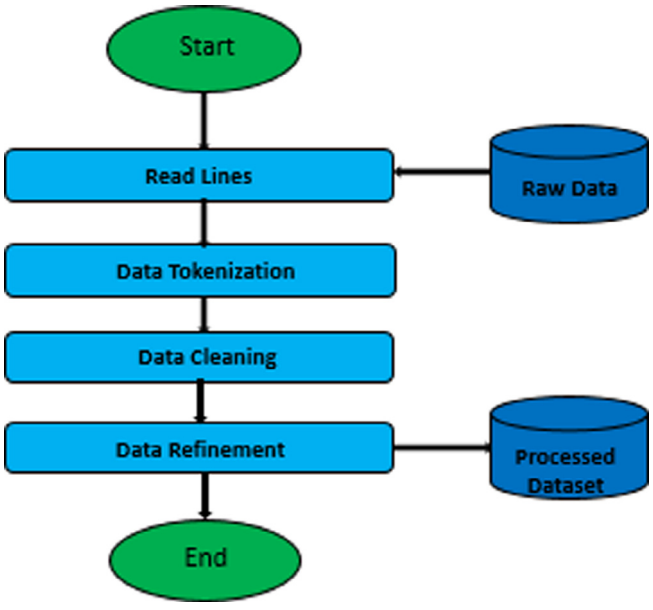

Fig. 3. The flowchart of data preprocessing and preparation steps.

Table 3
The total number of words extracted from processed files.

Corpus name	Num. of clean words
Al-Watan-2004 Corpus	85,052
Al-Jazeera Corpus	1,156,428
Al-Khaleej-2004 Corpus	2,272,750
Al-Watan-2004 Corpus	9,226,283
Books Corpus	74,770
KACST Corpus	2,036,728
Quran	66,314
RDI	74,959
Tweets	234,326

4.2. System implementation

The proposed methodology was experimented using a Web-based application. For the Web application development, it is implemented using Oracle APEX *19.1*. APEX is a rapid development framework from Oracle, and it is used to have a user-friendly interface through which learners can interact, register, and take the test. The system administrator can use this interface to analyze test results and other stored data as well as the ability to create relevant reports and dashboards. An Oracle APEX workspace application had been created. This workplace had accomplished the task of creating the automated version of the LRT test. Oracle SQL

Table 4
Summary of real words and nonwords datasets.

Item	Ave. word length	Count
Clean Dataset	6.5	14,000,849
Main Dataset-Distinct	6.5	399,495
Nonwords	5.2	38,412,714

statements were created to select and manipulate data of test tables in the database schema, some conditions were used in the query like frequency, length, and type (orthographic and phonological) to tune test difficulty and flavor. In the test window, some personal learner details are requested such as native language, age, gender, and number of years of learning Arabic. These meta data and the results of the given test will be used to analyze the test's dataset. To generate the human-crafted nonwords, we got help from an Arabic expert, who teaches Arabic language for many years. This expert had applied same rules that were used to generate orthographical and phonological nonwords in the automated approach and used the same rules to manually generate nonword types. This part is intended to hold a comparative study between automatic generated test and the manual generated one for validation purposes.

To create test tables, LRT items were considered; 20 questions were selected having 1:2 (word: nonword) ratio; and an SQL query was used to retrieve three equal parts (real word, orthographic, phonological) in random way based on parent word frequency; word length is adjusted to be between 4 and 9 characters to be within the range length of the collected data. Tables 5 and 6 summarize the main findings derived from these experiments with some samples. We divided the analysis into six categories based on the nonword generation type as follows:

- Both-Auto: Automatic generation of nonwords, *i.e.,* replacement a character with one of its orthographic and phonological similarity lists.
- Both-Manual: Manual generation of nonwords, *i.e.,* replacement a character with one of its orthographic and phonological similarity lists.
- Orthographic-Auto: Automatic generation of nonwords, *i.e.,* replacement a character with one of its orthographic similarity lists.
- Orthographic-Manual: Manual generation of nonwords, *i.e.,* replacement a character with one of its orthographic similarity lists.
- Phonological-Auto: Automatic generation of nonwords, *i.e.,* replacement a character with one of its phonological similarity lists.
- Phonological-Manual: Manual generation of nonwords, *i.e.,* replacement a character with one of its phonological similarity lists.

Table 5
Summary of the number of correct answers/ generation type and the overall accuracy.

Generation Type	Description	# of correct answers	Accuracy %
Both-Auto	Automatic nonword generation using orthographic and phonological similarity lists per letter	78	5.31
Both- Manual	Human nonword generation using orthographic and phonological similarity lists per letter	107	7.28
Orthographic-Auto	Automatic nonword generation using orthographic similarity list per letter	109	7.41
Orthographic-Manual	Human nonword generation using orthographic similarity list per letter	64	4.35
Phonological-Auto	Automatic nonword generation using phonologic similarity list per letter	172	11.7
Phonological-Manual	Human nonword generation using phonological similarity list per letter	173	11.77
Real word	Concrete words taken from the processed corpus	257	17.48
Sum = 1470		**960**	**65%**

4.3. Evaluation measures

Several generic evaluation measures were used to evaluate the performance of the suggested approach. For this work, we focus on the most common ones, specifically we consider accuracy, precision, and recall. The first three measures can be computed with the help of confusion matrix as shown in Fig. 4. To understand this figure, we provide the following definitions that are based on the work by Hamed (2019).

- True Positive (TP) is the number of correct answers, *i.e.,* positive class correctly identified as positive (real words that are identified as real words).
- True Negative (TN) is the number of correct answers, *i.e.,* negative class correctly identified as negative

  (nonwords that are identified as nonwords).

- False positive (FP) is the number of incorrect answers, *i.e.,* negative class incorrectly identified as positive (nonwords that are identified as real words).
- False Negative (FN) is the number of incorrect answers, *i.e.,* positive class incorrectly identified as negative (real words that are identified as nonwords).

Fig. 4 displays the confusion matrix of the learners' responses. The test had 30 input words, each one has 49 responses, so the total is 1470 observation, 490 out of 1470 is coming from answers to the real words input set (10 real words with 49 response for each word), 257 out of 490 observations had correct answers but 233 had incorrect answers. From another side, observations of answers come from nonwords are 980, 277 out of 980 were able to distract the learner by considering them as real words, while 704 out of 980 were not able to distract the learner and they had given nonwords selection. From the output of the confusion matrix, it is shown that 1/3 of the generated nonwords were able to distract the learners. Consequentially, the computed values of accuracy, recall and precision are 65%, 0.52 and 0.48, respectively. These small values indicate that the LRT questions that were automatically generated by the proposed system had confused the learners. It is correct that the frequency had enhanced the flexibility to determine test's difficulty, and this supports the idea of multi-level test generation, but the drawback is having less distinct count when comparing it with the total Arabic real words which is about 12 million words. This will affect the accuracy of the generated nonwords, while it is being classified as nonwords.

4.4. Comparative analysis

As clearly stated in the manuscript, Lexical Recognition Test (LRT) themes are one of the main methods that are widely used to measure language proficiency levels of some international languages such as English, Spanish, and German. However, similar research for Arabic is still at development stages, and existing proposals mainly use human-crafted methods for nonwords' generation. This research work aims at enriching Arabic LRTs, more precisely to generate nonwords in an automatic way, which can be considered as the first work to be conducted in this domain. Nevertheless, we tried to make a comparative study considering the most relevant contributions coming from Arabic and English. We specifically consider the work done by Hamed and Zesch (2018b) and Hamed (2019) who rely on human-crafted nonwords for Arabic LRTs, LexTALE (Lemhöfer and Broersma, 2012) and the high-order position specific language models (3-gram-ps) for the automatic generation of LexTALE-like English LRTs (Cavnar and

Table 6
A sample of nonwords using auto generation type.

Word	Type	New nonword	Old char	New char
آيات	automated_both	آياث	ت	ث
ادعهن	automated_both	ادغهن	ع	غ
رتبتها	auto_orthographic	رتبثها	ت	ث
غيبتهم	auto_orthographic	غيبيهم	ت	ب
موظفته	auto_orthographic	موظقته	ف	ق
تخذل	auto_phonotical	تخذق	ل	ق
سأنتهي	auto_phonotical	سأنتهل	ي	ل
صدقيني	auto_phonotical	صدكيني	ق	ك
مفاعل	auto_phonotical	مفاعط	ل	ط
ملامحا	auto_phonotical	ملومحا	م	و

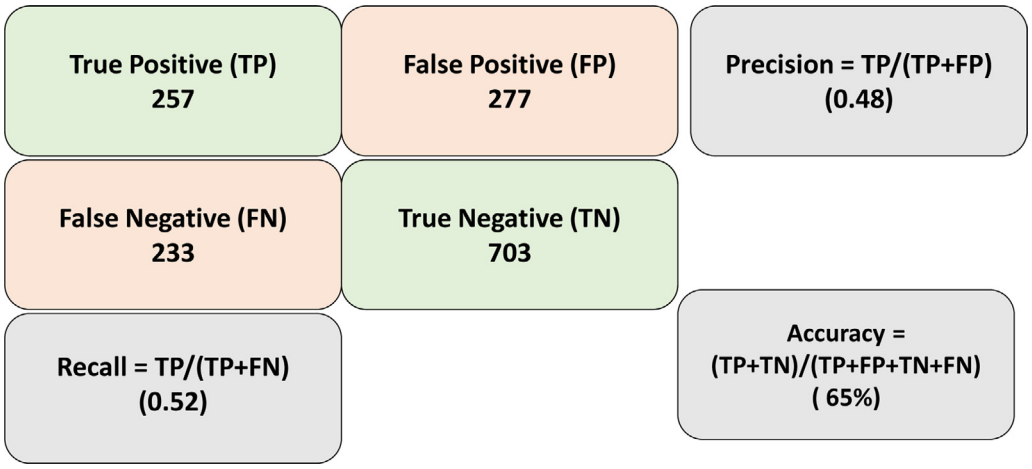

Fig. 4. The output of the confusion matrix and evaluation measures.

Trenkle, 1994; Hamed and Zesch, 2015). To ease the comparison, we considered the same evaluation measures used by these previous works. We mainly focus on recall and precision, which are two common matrices being widely used to measure performance of similar studies. In this comparative study, we considered recall and precision for words and nonwords calculations, R_w, R_{nw}, P_w, P_{nw}, respectively. They can be computed using the following equations (Fig. 4):

$R_w = TP / (TP + FN)$

$R_{nw} = TN / (TN + FP)$

$P_w = TP / (TP + FP)$

$P_{nw} = TN / (TN + FN)$

Table 7 summarizes the comparative results of recall and precision for words extraction and nonwords generation for two languages: English and Arabic. As the table shows, part of the data was obtained from the results reported by Hamed and Zesch (2018a), Hamed and Zesch (2018b) and Hamed (2019). It is worth noting that the Arabic script has diacritics, therefore results are divided into different columns. "ND" indicates that the test is non-diacritized. "Diac" indicates that the test is diacritized, whereas "Diac Freq." and "Diac InFreq." indicate that the test is diacritized using the frequent and infrequent diacritics.

The recall for words (R_w) refers to the portion of correct positive classifications (TP) from the cases that are positives. Whereas the Recall for nonwords (R_{nw}) refers to the portion of correct negative (TN) classifications from the cases that are negative. As per our approach, it can be clearly seen that the recall for words (R_w) – aka sensitivity, and nonwords (R_{nw}) – aka specificity – are the lowest among others, 0.52 and 0.72, respectively. This means that words' extraction and nonwords generation by the proposed approach have better quality and consider this as improvement over the past studies. For words' extraction, we argue that this is due to the lowest frequency of the tuning threshold parameter being used in the proposed algorithm (Algorithm 1). In other

Table 7
Comparative table showing differences between previous and proposed approach.

		English		Arabic					
		LexTALE	3-gram-ps	Hamed and Zesch (2018a), Hamed and Zesch (2018b)		Hamed (2019)			Proposed approach
				ND	Diac	ND	Diac $_{Freq.}$	Diac $_{InFreq.}$	
Recall	R_w	0.70	0.73	0.68	0.74	0.95	0.92	0.80	0.52
	R_{nw}	0.75	0.90	0.90	0.93	0.89	0.82	0.85	0.72
Precision	P_w	–	–	0.93	0.96	0.95	0.91	0.92	0.48
	P_{nw}	–	–	0.65	0.70	0.93	0.90	0.71	0.75

words, the selected words are used to belong to frequency classes that are less than the average frequency for all words in the corpus. Regarding nonwords, we argue that this is due to the confusion caused by similarity between the forms of Arabic letters. However, there is a room for improvement in generating Arabic nonwords.

Regarding to the precision which refers to the portion of words (P_w) and nonwords (P_{nw}) that correctly identified by the learners. As per the proposed approach, the calculated P_w and P_{nw} are 0.48 and 0.75, respectively. Those values are quite similar to the reported precision values. This means that learners have incorrectly identified the words and nonwords in 0.52 and 0.25 of the cases, respectively.

4.5. Analysis of LRT dimensions

The LRT was conducted in collaboration with local Arabic learning centers. The test was completed by 49 participants (15 female, 34 male) categorized into main groups: native and non-native speakers (foreign learners who joined some Arabic learning courses with different levels of learning years). In this section we discussed the learners' backgrounds using various dimensions and studied the relationships between various test components including word length, nonword generation type, learning years, and participants' main language. Pertaining to this analysis, we observed that 1200 responses were produced by native learners while 270 by foreign learners. Foreign learner's average age is 35 with average 5 years of learning journey.

A. *Word length:* Referring to Fig. 5(A) that illustrates the relationship between the word length and the correct score, we found that learners were able to distinguish between real words and nonwords when word length was in range of (4–6). This means that this range was less confusing than being in length of 3 or 7 characters. This implies that generating nonwords from a real word with length 3 or from a real word with length 7 or above, will produce high quality nonwords, which are difficult to be identified when comparing them with nonwords that had been generated from real words of length between 4 and 6 characters.
B. *Word type:* Referring to Table 5 that shows automatic generated types of nonwords and Fig. 5(B) that illustrates the relationship between word type and the number of correct answers, the real word part takes into considerations that most learners were native; on the other hand, phonological got the second score, and this is expected since phonological replaces Arabic letters based on phonetical considerations, and this might make the generated nonword sounds strange. From the opposite part, orthographical type and orthographical and phonological combination type had the minimum scores, and this is due to the argument that says if someone can identify the word, he/she knows it, and in the last cases learners had made identification because these nonwords were having high quality and it is difficult to judge. conclusions were derived: as expected, the highest scores were achieved when the learners answered.
C. *Learning years:* Fig. 5(C) depicts the relationship between the learner's language levels and the number of correct answers. It is concluded that the number of learning years could not be counted as a segregated item as the number of correct answers per learning years does not have a considerable variance among all learning levels.
D. *Learners' main language:* Fig. 5(D) displays the learners' responses distribution, where (1200 out of 1470) observations were produced by native learners, while (270)

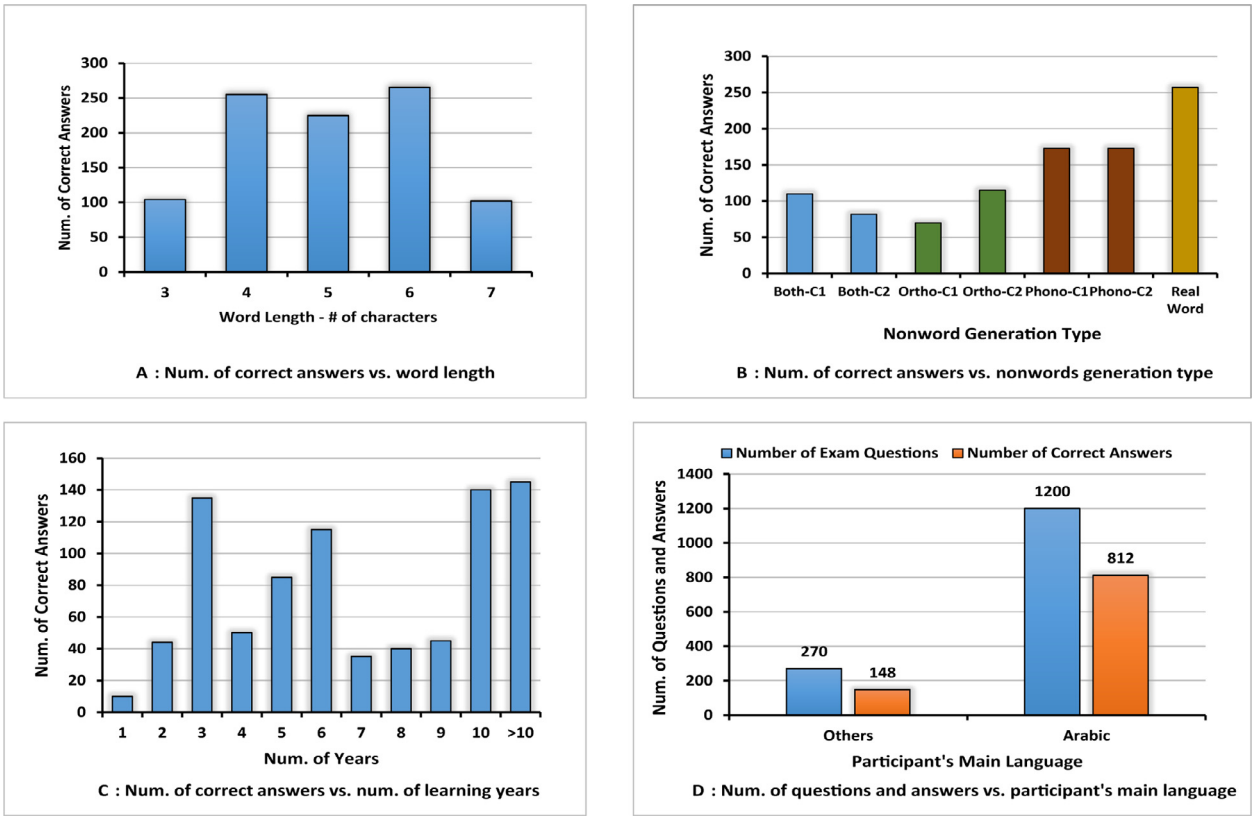

Fig. 5. Histograms that show the relationships between various exam dimensions (word length, nonword generation type, learning years, and participants' main language) and the number of correct answers.

observations were produced by nonnative learners. This high discrepancy explains why it is only one third of nonwords had confused the learners since native learners can identify their language better than non-native learners. Fig. 5(D) depicts that the total number of correct answers were (812 + 148, 65%) out of observations total operations, and (388 + 122, 35%). As shown in this figure, the most correct answers had been produced by native learners. Despite of this, we could consider that mother language is a segregator, since most of the participated learners were Arab. To judge this conflict, we need to have similar groups to conduct comparative studies with closed values of suggested learner dimensions.

E. *Gender Impact*: Fig. 5(D) show the relationships between the numbers of correct answers vs. participant's gender. Most correct answers were achieved by males regardless their mother tongue, but in spite of this, we could not prove that males could have higher scores than females since male participant's count is much more than female count. To prove that gender is a segregation dimension, we could have similar gender counts and similar dimensions as age, learning years, etc.

It is clearly shown that the observations fulfill expectations as we can find the main segregator dimension is the vocabulary generation type. In this study, robust nonwords that confused learners are the one which are generated based on the orthographic similarity rules, since this modification guarantees the vocabulary generation of much closed shapes of the original ones, that is why it was confusing. In addition to this observation, the highest quality of nonword generation that achieved the least score is the nonwords that have replacement letters based on orthographic and phonological similarities (check automated_both type in Table 5). In other words, when the replacement letter could be in the intersection set between orthographical and phonological similarity groups.

4.6. Results validation

A human-driven intervention is used as an extra verification step to validate the quality of auto generated nonwords. Our main argument here is that incorporating a manually generated vocabularies by Arabic expert following the same rules will enable us to double-check the types and quality of nonwords. To do that, we designed a sample version of the LRT following same LRT versions implemented in other languages. We prepared a test version with a ratio 1:2 (real words: nonwords), some questions were written by Arabic expert following the same rules. Thus, in this study, the created LRT contained 30 questions, 10 real words and 20 nonwords. The list of nonwords is divided into two groups (10 nonwords generated manually and the other 10 generated automatically) with possible answers (True, False) for each question. The purpose of this experiment is to make sure that the LRT vocabularies that are already taken from real corpus, auto generated nonwords, and those generated manually by Arabic experts have good quality that could distract learners.

A verification step was carried out to validate the above results. The main argument here is that generating good nonwords in Arabic is an eligible technique that can be used to establish Arabic LRT version. The results proved this argument as well. Besides this result, we could find that there is no contradiction between manual and automated methods, this indicates that automated nonwords generation is a valid option that could achieve results as the manual version prepared by a language expertise. For obtaining more accuracy, more focus was given on generating nonwords based on letter replacement with its corresponding set, with the intersection between orthographical and phonological similarity groups. This intersection intended to produce high quality nonwords, taking into considerations words frequency rank.

We also computed the p-value using T-Test, which indicates a statistical convergence of compared values of two selected logical groups of participants. To do that, we have randomly divided the participants into two groups: Group (A) and Group (B). After selecting the groups, we calculated the score for all participants. Then, the scores for both groups are compared using a paired T-Test. The results have shown that the two-tailed p-value is less than 0.0001, which is less than the 0.05 threshold used for this test. As per the conventional criteria, the scores are statistically significant.

4.7. Applications

To get a better picture on the practical value of the proposed approach, we chose three example applications: First, since LRT themes are internationally well-recognized methods to measure learning proficiency levels of common languages such as English, German, and Spanish, similar research for Arabic are still at development stages, and existing proposals mainly use human-generated methods. Therefore, one of the potential applications of the proposed approach is that it can be used to measure the proficiency level of Arabic learners (Arabic LRT). Second, another potential application is Arabic spellchecker. Since the proposed approach can generate huge amount of high quality nonwords, these nonwords can be stored in a database of vocabularies – it can be integrated into Arabic Proofreading tools – that can be used as a reference for spell checking documents written in Arabic language to ensure that any Arabic documents are accurate, readable, and meet professional standards. Third, since Arabic LRT is still at development stages, the proposed approach can be considered by many researchers who conduct relevant studies. The project's source code, its implementation steps, documentations, and the generated nonwords database will be freely available on Github platform (https://github.com/).

5. Conclusion and future work

This paper suggested a new methodology, based on a newly developed algorithm, to automatically generate high quality nonwords to act as stimuli in Arabic LRTs. The new algorithm generates nonwords based on Arabic character classifications theory: phonological, orthographic, n-grams, and word frequency. The developed method was tuned using n-grams and word frequency map to design multilevel LRT. The results have shown that the proposed approach is mature enough for generating high quality and confusing nonwords. From the output of the confusion matrix, 1/3 of the generated nonwords have received wrong answers by the leaners. This hypothesis had been approved when results of manual generated nonwords have similar results of the automated method and both types confused the learners.

As a future work, we will plan to include Arabic words diacritization to enhance the test and make the generated words more robust. Other future works for further improvements could be using Arabic poets in traditional Arabic, religious sources and traditional Arabic transcripts and literature. Also, some further improvements on the test setup shall be considered such as similar groups with singular words, similar groups with plural words, similar groups with mix words, test versions with only one type of word category (names, verbs, adverbs and plural).

References

Abdelgadir, E.M., Ramana, V.L., 2017. A Handbook on Introduction to Phonetics and Phonology: For Arabic Students. Notion Press.

Abdul-Mageed, M., 2017. Not all segments are created equal: Syntactically motivated sentiment analysis in lexical space. Proceedings of the 3rd Arabic natural language processing workshop (WANLP2017).

Al-Twairesh, N., Al-Matham, R., Madi, N., Almugren, N., Al-Aljmi, A.-H., Alshalan, S., Alshalan, R., Alrumayyan, N., Al-Manea, S., Bawazeer, S., Al-Mutlaq, N., Almanea, N., Huwaymil, W.B., Alqusair, D., Alotaibi, R., Al-Senaydi, S., Alfutamani, A., 2018. Suar: Towards building a corpus for the Saudi dialect. Procedia Comput. Sci. 142, 72–82.

Balota, D.A., Yap, M.J., Hutchison, K.A., Cortese, M.J., Kessler, B., Loftis, B., Neely, J.H., Nelson, D.L., Simpson, G.B., Treiman, R., 2007. The English lexicon project. Behav. Res. Methods 39 (3), 445–459.

Bougrine, S., Chorana, A., Lakhdari, A., Cherroun, H., 2017. Toward a Web-based speech corpus for Algerian dialectal Arabic varieties. In: Proceedings of the 3rd Arabic Natural Language Processing Workshop, pp. 138–146.

Cavnar, W.B., Trenkle, J.M., 1994. N-gram based text categorization. In Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval, 48113(2): 161–175.

Duyck, W., Desmet, T., Verbeke, L.P.C., Brysbaert, M., 2004. WordGen: a tool for word selection and nonword generation in Dutch, English, German, and French. Behavior Res. Methods, Instrum. Comp. 36 (3), 488–499.

Elfardy, H., Diab, M., 2012. Aida: Automatic identification and glossing of dialectal Arabic. In: Proceedings of the 16th EAMT conference (project papers, pp. 83–83.

Ellis, N., 2002. Frequency effects in language processing: a review with implications for theories of implicit and explicit language acquisition. Stud. Second Language Acquisit. 24 (2), 143–188. https://doi.org/10.1017/S0272263102002024.

Farghaly, A., Shaalan, K., 2009. Arabic natural language processing: Challenges and solutions. ACM Trans. Asian Language Inf. Process. (TALIP) 8 (4), 1–22.

Gueddah, H., Yousfi, A., 2013. The impact of Arabic inter-character proximity and similarity on spell-checking. In: Proceedings of the 8th International Conference on Intelligent Systems: Theories and Applications (SITA), Rabat, 2013, pp. 1-4, doi: 10.1109/SITA.2013.6560811.

Guellil, I., Saâdane, H., Azouaou, F., Gueni, B., Nouvel, D., 2019. Arabic natural language processing: an overview. J. King Saud Univ.-Comp. Inf, Sci.

Habash, N.Y., 2010. Introduction to Arabic natural language processing. Synthesis Lect. Hum. Language Technol. 3 (1), 1–187.

Habash, N.Y., 2017. Language Technologies for Arabic and Its Dialects ArabWIC. New York University Abu Dhabi, Beirut.

Hamed, O., 2019. Automatic Generation of Lexical Recognition Tests Using Natural Language processing Doctoral dissertation. Universität Duisburg-Essen).

Hamed, O., Zesch, T., 2018b. The role of diacritics in adapting the difficulty of Arabic lexical recognition tests. NLP for Computer Assisted Language Learning (NLP4CALL 2018), 23.

Hamed, O., Zesch, T., 2015. Generating nonwords for vocabulary proficiency testing. In: Proceeding of the 7th Language and Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics, pp. 473–477.

Hamed, O., Zesch, T., 2017. The role of diacritics in designing lexical recognition tests for Arabic. Procedia Comput. Sci. 117, 119–128.

Hamed, O., Zesch, T., 2018a. Exploring the effects of diacritization on Arabic frequency counts. In: 2018 2nd International Conference on Natural Language and Speech Processing (ICNLSP). IEEE, pp. 1–6.

Hegazi, M.O., 2016. An approach for Arabic root generating and lexicon development. IJCSNS Int. J. Comp. Sci. Network Security 16 (1).

Huibregtse, I., Admiraal, W., Meara, P., 2002. Scores on a yes-no vocabulary test: correction for guessing and response style. Language Test. 19 (3), 227–245. https://doi.org/10.1191/0265532202lt229oa.

Jarrar, M., Habash, N., Alrimawi, F., Akra, D., Zalmout, N., 2017. Curras: an annotated corpus for the Palestinian Arabic dialect. Language Resour. Eval. 51 (3), 745–775.

Keuleers, E., Brysbaert, M., 2010. Wuggy: a multilingual pseudoword generator. Behav. Res. Methods 42 (3), 627–633.

Khalil, A., Darwish, A.A., 1967. Al-'Ayn: First Arabic dictionary by al-Khalil ben Ahmad al-Farahidy. Baghdad: Matba'at al-'Ani.

Lemhöfer, K., Broersma, M., 2012. Introducing LexTALE: A quick and valid lexical test for advanced learners of English. Behavior Res. Methods 44 (2), 325–343.

Menacer, M., Mella, O., Fohr, D., Jouvet, D., Langlois, D., Smaili, K., 2017. An enhanced automatic speech recognition system for Arabic. In: Proceedings of the 3rd Arabic Natural Language Processing Workshop – EACL 2017

Nadkarni, P.M., Ohno-Machado, L., Chapman, W.W., 2011. Natural language processing: an introduction. J. Am. Med. Inform. Assoc. 18 (5), 544–551.

Rastle, K., Harrington, J., Coltheart, M., 2002. 358,534 nonwords: the ARC nonword database. Quarterly J. Exp. Psychol. Section A 55 (4), 1339–1362.

Ricks, R., 2015. The development of frequency-based assessments of vocabulary breadth and depth for L2 Arabic. Georgetown University-Graduate School of Arts & Sciences.

Salloum, W., Habash, N., 2014. ADAM: Analyzer for dialectal Arabic morphology. J. King Saud Univ.-Comp. Inf. Sci. 26 (4), 372–378.