Intelligent Information Management, 2019, 11, 1-23
http://www.scirp.org/journal/iim
ISSN Online: 2160-5920
ISSN Print: 2160-5912

Developing Two Different Novel Techniques for Arabic Text Stemming

Mohammad Mustafa[1], Afag Salah Aldeen[2], Mohammed E. Zidan[3], Rihab E. Ahmed[4], Yasir Eltigani[5]

[1]Department of Computer Information Systems, Faculty of Computers and Information Technology, University of Tabuk, Tabuk, Saudi Arabia
[2]Department of Computer Science, College of Computer Science and Information Technology, Sudan University of Science and Technology, Khartoum, Sudan
[3]Department of Mathematics, Faculty of Science, University of Tabuk, Tabuk, Saudi Arabia
[4]Department of Computer Information Technology, Faculty of Computers and Information Technology, University of Tabuk, Tabuk, Saudi Arabia
[5]Department of Computer Information Systems, Ahmed Bin Mohammed College, Doha, Qatar
Email: mmustafa@ut.edu.sa, afagsalah@hotmail.com, shoshoza@gmail.com, r.mussa@ut.edu.sa, yeali@hotmail.com

How to cite this paper: Mustafa, M., Aldeen, A.S., Zidan, M.E., Ahmed, R.E. and Eltigani, Y. (2019) Developing Two Different Novel Techniques for Arabic Text Stemming. *Intelligent Information Management*, **11**, 1-23.
https://doi.org/10.4236/iim.2019.111001

Received: October 17, 2018
Accepted: December 31, 2018
Published: January 3, 2019

Abstract

Stemming is used to produce stem or root of words. The process is vital to different research fields such as text mining, sentiment analysis, and text categorization, etc. Several techniques have been proposed to stemming Arabic text and among them, Khoja and light-10 stemmers are the most widely used. In this paper, we propose and evaluate two different stemming techniques to Arabic that are based on light stemming techniques. The new stemmers are compared to best reported light stemmer, which is light-10. Results and experiments, which were conducted using standard collections, reveal that The proposed stemmers yield 5.13% and 13.1% improvement in retrieval performance over light 10 with 0.369 average precision and 0.397, respectively and the improvement is statistically significant.

Keywords

Arabic Language, Arabic Information Retrieval, Light Stemming, Light 10, Extended Light-Stemmer, Linguistic-Based Stemmer

1. Introduction

Arabic Language is the largest group of Semitic languages. It is the native language for more than four hundred millions [1] centered in the Arabic region, which includes North Africa and Middle East countries. As in the majority of the

Semitic languages, Arabic language is written from left to right and its script has 28 letters. However, unlike the popular Semitic languages, words are often written in a cursive (non-concatenative), rather than discontinuous, longhand style [2] but with spaces to delimit words from each others. Diacritics and vowels are usually omitted in Arabic script. As a result for this cursive style each Arabic letter can be written in different glyphs according to its position in words, e.g. عـ, ع (the Arabic letter ع). Letters do not have different upper or lower cases.

In contemporary time, the term Arabic has three forms [3] [4]: Classical Arabic (CA), Modern Standard Arabic (MSA) and Dialectal Arabic. Classical Arabic was the language of old Arabic-speaking people, e.g. pre-Islamic times and during the appearance and rise of Islam. A typical example for classical Arabic is exemplified in the Holy Quran. Modern Standard Arabic, known also as Fusha, is a modified version—with a modern vocabulary—of Classical Arabic. It is typically found in news papers and includes many technical terms that are not originated in the language. For instance, words like كمبيوتر (meaning: computer), and فيديو (meaning: video), etc., are not Arabic words. MSA is also used in official speech and communication and it is the formal language of the media and education across the Arabic world. Dialectal Arabic, as the name indicates, is used in informal communication in all Arabic-speaking countries and its vocabulary is regionally variant. Due to this classification, the term "Arabic" refers to both MSA and Dialectical Arabic [3] [5].

As in Semitic languages, root in Arabic is often tri-literals, quad-literals or pent-literals with five consonants. Arabic has 10,000 different roots [6] and about around 1200 of them are only used in official MSA [7]. Arabic words are composed by adding affixes (*i.e.* antefixes, prefixes, suffixes) to these roots, resulting in a large number of possible words in Arabic for each root. Affixes include definite articles, conjunctions, particles, and other prefixes, whereas suffixes may include dual feminine and plural masculine. Words in Arabic are either masculine or feminine. It may also, meaning Arabic suffix, include postfixes which are used to indicate pronouns (*i.e.* second and third person). For example, the word لننصرنهم (meaning: we will surely support them) can be decomposed as follows: (antefix: ل, prefix: ن, root: نصر, suffix: ن and postfix: هم). This identified attribute in Semitic languages in general, and in Arabic particularly, makes many Arabic words appear only once in texts compared to Indo-European Languages [5] [8].

The major pattern from which the majority of the Arabic words derived, is the pattern فعَل (transliterated as f-à-l), which correspond to tri-literal roots. Patterns also can be affixed by adding letters at the beginning, medially or at the end, resulting in more regular patterns. Usually words in Arabic are formed according to these patterns.

As a number of words can be formed of a single root or pattern, the opposite process, which is known as Stemming, is the task of rendering all the conflated forms of a word into a single form known as stem. For instance, from the three

consonants trilateral root زرع (meaning: to farm), several words can be formulated such as: مزروع (meaning: farmed), مزارع (meaning: farmer), مزارعة (for singular feminine in nominative, accusative and genitive cases), مزارعان (for dual masculine in nominative case), مزرعة (meaning: farm), etc. Thus, the stemming process tends to cluster all forms of a single word into a single stem and hence, these forms can be handled as synonymous words that have the same meaning or concept. Identifying synonymy could have a significant impact on finding the most relevant information. Therefore, stemming is a vital process for several fields including information retrieval, sentiment analysis, text mining, text categorization and classification, etc.

On the other hand, stemming may erroneously group words with different meanings and concepts into a single stem. For instance, Consider the words سيار (meaning: cell-phone), مسار (meaning: path), مسرة and (meaning: pleasure). In spite of the different meaning of each of these words, all of them are formulated from the trilateral root سار (meaning: walk). This feature concluded that Arabic stemming is not a straight process and could hurt performance in some cases.

A large number of studies have explored different techniques for Arabic stemming. However, the major two approaches are heavy stemming (known also as root-based stemming) and light stemming. Heavy stemming always tries to pull out the stem or the root from the input word. For instance, the root, which is the singular third person in perfective (past) form in the Arabic word الحاسوب (meaning: computer) is حسب. On the other hand, light stemming attempts to stem the input word lightly by stripping off affixes (*i.e.* prefixes and suffixes) only, resulting in what is known as stem. The stem is the minimal canonical form of words after removing the majority of prefixes and suffixes, e.g. the stem of the word الحاسوب (meaning: computer) is حاسب.

This paper proposed two different stemming techniques to Arabic. Motivated by the reported results in the literature, in which light stemming is the best known approach to Arabic, the first approach is a new light stemmer that has been developed on the top of the best reported light stemmer in Arabic review. The work is different from best known approach to stemming in two points. First, it introduces more prefixes and suffixes, in particular, those usually attached to verbs. This is because during our analysis, we noticed that the best known stemmer relatively neglects some of the affixes that are commonly used in verbs. One might look to truncated prefixes and suffixes in light-10 and could be easily recognized that the removed affixes are focused around nouns rather than verbs. On the other hand, our algorithm applies different heuristic rules to ensure that affixes that are parts of words, will not be eliminated.

Following this, a new linguistic stemmer has been also proposed. The method is inspired by a strong principle in Arabic grammar which states that different patterns are used for different part of speech of words. Thus, we believe that the process of stemming should not be dependent on a specific approach. However, in spite of the implementation of such a similar approach in a few number of

studies, but our work is different. First, in our approach two different predefined lists (one for verbs and the other for nouns) were used. In all other studies, only a single list of patterns that did not distinguish between verbs patterns and nouns patterns, were built. Using only a single list is an invalid assumption because many verbs and nouns may share the same pattern. Second, we make use of a POS tagger. However, in order to avoid putting some burden during stemming on IR system due to tagging, we only use the POS when the preceding classification phases fail.

The concluded results reveal that the proposed two stemmers yield significant improvement over the best known light stemming approach, which is light 10 and the difference is statistically significant.

2. Related Work

Different methods and approaches have been investigated to analyze Arabic stemming problem. The majority of these studies are dedicated to the issue of which is the best term to index Arabic text: is it the stem or the root. However, a considerable number of the reported papers concluded that the effectiveness of stem-based methods is much better than those based on roots. In fact, most of the proposed approaches claim that higher accuracy is achieved. The used datasets in these reported studies are very varied also and the majority of the employed test collections are not standard. In [3] the authors of this paper provide a complete survey for the developed algorithms of Arabic IR.

A considerable number of the developed algorithms for root-based stemming rely mainly on the removal of affixes either using some dictionary (table lookup and/or list-driven patterns); or developing a set of linguistic rules to recognize verb patterns and consequently extracting root. In the former technique words are stored with all its possible decompositions. Thus, the input word is analyzed to determine the best decomposition and thus the root is retained. On the other hand, linguistic rules often take the input word and attempt to remove its prefixes and suffixes after matching them with a pre-stored list of affixes. The left part of the word is often gone through exhaustive analysis and matched against some pre-listed patterns so as to pull out the root.

Khoja and Garside [9] and Buckwalter [10] stemmers are examples for heavy stemming algorithms, which attempts to pull out the root of the input Arabic words. Both stemmers rely on using dictionary. Khoja depends on some stored predefined patterns and list-driven roots. This is done after removing the longest prefixes and suffixes that match. Buckwalter depends on the use of some stem tables that include prefixes, possible stems and suffixes. These tables guide the algorithm to produce possible combinations (prefixes, stems and suffixes) of the input word and thus, if the combination is correct, then the stem (or all the possible stems) is obtained. Due to this mechanism Buckwalter may generate more than one stem. Both of the two stemmers were widely used in Arabic IR. In particular, Khoja stemmer is the most cited work in Arabic IR.

Sebawai, which is a root-based analyzer, provided by Darwish [11] follows a similar approach with one major difference that the stemmer computes earlier the likelihood of occurrence of each (prefix, suffix and stem) template according to an automatically generated dictionary of 'word-root' pairs. So when a word is to be stemmed, a probability for each possible combination of (prefix, suffix and stem template) is computed and the stem with the higher probability is chosen. Results revealed that the stemmer has some weaknesses when it attempts to handle transliterated named entities.

Xu, *et al.*, [12] built their stemmer on the top of Buckwalter analyzer but, with a major difference that if more than one stem is returned by the algorithm, then those stems are handled as equally probable and then the use of the probabilistic IR manages that ambiguity. The reported experiments in this work concluded that stemming in such a way outperformed full-word stem.

Inspired by Khoja, AlSughaiyer and Alkharashi [13] proposed producing word patterns from root, e.g. the pattern فعال can be applied to the root قتل (meaning: to kill) and then verb pattern is compared to words. The algorithm was promising and simple.

Shalabi, *et al.*, [14] in their root-based stemmer, proposed to extract root and patterns based on excessive letter positions. Therefore, in the study, all patterns and Arabic letters with their positions in words were stored into dictionaries. During stemming some rules are applied based on these dictionaries to extract roots. The authors claimed that 95% accuracy is achieved but only few words were chosen to test the algorithm.

Contrarily to root-based stemmers, light stemmers have been also implemented. Unlike root-based techniques, which employ Arabic rules to extract roots, light stemmers attempt to remove the most frequent prefixes and suffixes with or without using some Arabic corpus [15] [16] [17] and [18]. The major difficulty here is that if a prefix or a suffix found, the decision of the removal of these affixes should be taken after applying some rules in order to avoid removing an affix which is a part of the word under stemming.

In their CLIR (Cross-Langauge Information Retrieval) experiments, Darwish and Oard [15] implemented a brute removal of the most common suffixes and prefixes but with no particular rules when these affixes are to be removed. Results showed that such removal could hurt performance as many prefixes may be original parts of the word to be stemmed.

Aljlayl and Frieder [16] work also stems Arabic words lightly but, it is different from previous work in that the developers managed the cases in which an affix is a part of word. This is accomplished by checking the numbers of letters left in the word. In the study, the authors applied some useful Arabic rules such as extending the shadda as it represents duplication of Arabic letter consonants. They also attempt to manage arabicized words. Arabicization is the process of writing words from other languages into Arabic letters, e.g. كمبيوتر (meaning: computer). Results in this study showed how useful is light stemming in Arabic

and the approach outperforms root-based algorithms, Khoja stemmer in particular.

Parallel corpora, which contain several monolingual sub-collections in different languages, have also been explored [18]. The approach is slightly different from previous studies because it depend solely on a basic assumption that Arabic words with the same root and semantic will be translated to a single English word. Accordingly, the idea is based on clustering Arabic words into a single cluster after stemming their peer English texts using an English stemmer. Following this, the shortest Arabic word is chosen. For the translation task, the authors used an online Machine Translation (MT) system. In the same study, the authors also developed a light stemmer which strips off the most common prefixes and suffixes and the removal is based on some heuristic rules. Results showed that light stemmer is better than clustering-based stemmer.

Nwesri, *et al.*, [19] developed a set of rules to strip off prefixes and suffixes from words under stemming before stemming them lightly. Their techniques for removing prefixes and suffixes and their stemmers are based on very restricted Arabic rules. The techniques are novel and reported good performance.

Larkey, *et al.*, [20] developed a set of light stemmers. The most widely used in the set is light 10 and each stemmer is different from others in the total number of prefixes and suffixes that are to be removed. In fact, prefixes and suffixes were accumulated at light 10. In light-10, Larkey suggests to remove lightly the prefixes (و ,لل ,فال ,كال ,بال ,وال ,ال) and the suffixes (ي ,ة ,هـ ,يه ,ين ,ون ,ات ,ان ,ها). However, in order to prevent the removal of affixes that are parts of the words' roots (and thus should be kept), the algorithm is handled with three rules: 1) remove the letter و (meaning: and) from the beginning of words if there are 3 or more remaining letters after removing the و (this condition is stated to prevent removing words that begin with the letter و 2) drop definite articles if this leaves 2 letters or more 3) remove suffixes, starting from right to left, from the end of words if this leaves 2 letters or more. In Larkey's study, light 10 was compared to two modified stemmers of the Buckwalter and Diab analyzers [21] and the reported experiments showed that light1-0 yields a better performance than both Buckwalter and Diab analyzers and the differences are statistically significant.

Inspired by both Buckwalter and light stemming algorithms, a combination technique for stemming Arabic words has been proposed by Kadri and Nie [17]. The researchers used a TREC (Text Retrieval Conference) corpus to decompose each word presented in its possible stems. During stemming, the algorithm uses that corpus statistics to choose the most appropriate stem. Results reported show that the algorithm has an advantageous feature over traditional light stemming technique in that it is able to determine semantic of words, which is the major lack of light stemming.

Inspired by the fact that light stemming may remove letters that are integral parts form Arabic words, Ababneh, *et al.* [22] proposes to match each word with a set of predefined Arabic patterns. If there is a matched pattern, the stem is re-

trieved otherwise the word is analyzed to its possible prefixes and suffixes and the stem is produced according to some compatibly list; and to whether the composition is valid. If the produced stem is invalid, the authors developed a set of rules based on Arabic grammar to count words lengths before removing affixes. The algorithm has been tested with very few numbers of words and thus, the results cannot be verified.

Inspired by that words with the same root and meaning in documents have the tendency to co-occur together in documents with the same topics, statistical-based stemmers have been used in the literature [23] [24]. The major aim behind this assumption is to prevent clustering words with the same stem (but have different semantic meanings) into a single clustered stem. For example both أطفال (meaning: children) and طفيليات (meaning: parasites) would be conflated to the single stem طفل though both words are semantically different. In order to extract such a strong feature, different similarity and association measures have been used. Examples include Dice Coefficient, Mutual Information, etc. Xu and Croft [24], and Larkey, *et al.*, [23] concluded that the technique is found to be promising but it does not outperform light-10.

Similarity measures have been also used for measuring similarity between n-grams of document words with n-grams of user query. Mustafa and Al-Radaideh [25] reported that the use of di-grams is better the using tri-grams but, the richness of the Arabic language makes the use of such approach is not a good option for indexing. Nevertheless, Xu, *et al.*, [26] stated contradictory results in which tri-grams are found to be better than bi-grams. This contradiction is mainly caused by the dataset that have been used in the two studies. In particular, the dataset used in Mustafa and Al-Radaideh was extremely small compared with the standard dataset that was used by Xu and his colleagues.

A similar technique has been also used by Hmeidi, *et al.*, [27] who tested both Dice and Manhattan coefficients for measuring similarity between bi-grams of words of documents and queries. The experiments, which were conducted using the Holy Quran, revealed that Dice distance coefficient outperforms Manhattan coefficient.

Al-Shammari and Lin [28] proposed a novel approach that is based on a simple assumption: It would be a good option to use light stemming for Arabic nouns while grouping verbs into a single root (meaning the use of root-based stemmer) is the best alternative for stemming verbs. Al-Shammari and Lin employed Arabic stopwords, which are indicators for the successor words tag (*i.e.* a verb or a noun), to classify verbs from nouns and then the best approach is used is used for stemming. The same trend was also followed by Mansour *et al.* [29]. However, both studies employed extremely small collections (not more than 57 documents) for testing the algorithms.

Artificial intelligence techniques such as Genetic Algorithms (GA) [30] and Back-Propagation Neural Network (BPNN) with multi-class classification [31] have been also investigated. For example, Alserhan and Ayesh trained their

Back-BPNN with 250 words with their correct roots. There were four classes in their study that represent Arabic frequent affixes. In the study, the accuracy of the developed network was found to be 84%. However, it is notable in the study that the dataset was extremely small (only 1000 words were used) and the word length does not exceed 4 letters, which is not optimal to Arabic rules for constructing roots.

To sum up our talk in this part, it is concluded in the literature that light stemming approaches are better than heaving stemming techniques. In fact, light stemming approaches are the most dominant among the existing approaches for stemming Arabic. But, each of the two paradigms has some pros and cons. On one hand, heavy stemming often results in over-stemming, leading to a low precision. Another major problem as discussed above is the fact that it is not always correct to produce the root of proper nouns or nouns in general. Let's consider the following nouns: المهرجان, المكاني, السودان, باراك أوبام and الستائر (meanings respectively: the republic of the Sudan, the festival, spatial, the US leader Barak Obama). Using a root based stemmer like Khoja, the stems are برا وبا, هرج, كني, سود and ستر, respectively. All of the stems are either chaotic or/and do not have similar semantic meanings to their original words.

On the other hand, light stemming preserves the meaning of words, unlike root-based techniques, and achieves the goal of retrieving the most pertinent documents, but it may not succeed to cluster semantically similar words together (under-stemming), resulting in low recall. Nevertheless, the majority of the studies devoted to stemming Arabic in IR reported that light 10 is the best known algorithm for indexing Arabic words and it has been identified as a fashionable solution to Arabic stemming. Light 10 has been added to the most famous IR systems like the Lucene and the Lemur toolkit. In his study to compare nine different Arabic stemmers (including Aljlayl and Frieder, Berkeley Team stemmer and Kadri's linguistic based stemmer, for examples), Eldesouki, *et al.*, [32], reported that light 10 outperforms all the listed stemmers and the difference in the majority of the comparisons was statistically significant. The same arguments were also concluded by the developers of light 10 [20] who stated that light 10 is far better than Khoja. In the same study the developers also reported that light 10 outperforms both Buckwalter and Diab analyzers [21].

Stemming techniques that use the idea of simple tagging seems elegant and effective but, one major weakness in existing approaches is that they depend solely on a few entries in a dictionary-driven approach. For instance, Al-Shammari and Lin [28] used only 2200 stopwords to classify verbs from nouns. This assumption may be valid for classifying only few Arabic words in large corpora, which are often used in IR systems. An explanation for this fact is that the majority of the Arabic words cannot be determined by only preceded words. This is may be the major reason for using only small text collections for experimenting the approaches in both Al-Shammari and Lin [28] and Mansour *et al.* [29] studies.

3. Proposed Stemming Approach

Given the above trends, this paper proposes two different stemmers that could minimize the major problems introduced in existing approaches. The first stemmer, which has been called Extended-Light, is a light stemmer that aims to suppress the impact of the under-stemming problem. This is done by introducing more prefixes and suffixes so as include clitics and those affixes that usually attached to verbs. The stemmer has been built on the top of the best reported light stemmer, which is light-10. Thus, Extended-Light can be applied independently to any Arabic texts. The second stemmer is a linguistic stemmer. It is motivated by a principle insight in Arabic morphology, which declares that words in Arabic are often rhymed into different patterns according to some different rules. Thus, the proposed linguistic stemmer employs these patterns for determining the correct part of speech of words and thus, choosing which stemming technique is to be used. The next section describes the proposed stemmers in more details.

3.1. Extended-Light Stemmer

It is known that Arabic prefixation system consisting of definite articles (like ال, which means the), prepositions (like the letter ب , which is pronounced as BAA), clitics (like the letter ف, which is pronounced as FAA) or a hybrid style between them (as in بالـ, which means with the) and the suffixation system containing pronouns (absence , person or possessive pronouns), dual and plural feminine (as in أستاذات, which means teachers) and dual and plural masculine (as in أستاذان, which means two teachers). For instance, a word like السودانيون (meaning: Sudanese) has been formed by adding the definite prefix (ال) and the plural masculine suffix (يون), resulting in السودانيون. Using this assumption, the problem of stemming in Arabic, has been changed to which prefixes and suffixes should be stripped off Arabic words and under what conditions those affixes should be truncated.

It was also noticed that most strippable prefixes and suffixes in light 10 are mainly focused around nouns. For instance, the prefixes of light 10, which are فال, كال, بال, وال, ال, وبال, ولل, are all determined for the purpose of removing these prefixes from grammatical nouns and proper nouns. Thus, in a word like السودان only the define article ال will be removed, resulting in سودان, while in the name of the American President باراك أوباما the name will be preserved as it was submitted to an IR system. Although the argument here indicates that this is a good feature can be accounted to light 10 stemmer but, in contrast it is the major reason for the under-stemming problem, in which words with the same meanings may be clustered into different groups, as it was described earlier.

At first, we did a very deep analysis to identify the types of problems that may occur when using light 10 stemmer, besides those already discussed. Several snippet codes were written for this purpose and for attempting to extract which prefixes and suffixes would be able to suppress some of the drawbacks that are

found on the best known light stemming approach. As a result for this analysis, it was concluded that the set of the stated affixes in light 10 stemmer is not enough to perform the best stemming technique. In particular, when we tackle light 10 behaviour, it was found that there are many prefixes and suffixes related to nouns that were not included in the stemmer although Arabic nouns represented a considerable part of the words in the language. Note that only few patterns can be used for verbs in Arabic while there is a lot of patterns that can be used for nouns. This is not a trivial notice as the ignorance of some of the major prefixes and suffixes that are related to nouns may easily cause the IR system to miss the documents and degrades retrieval effectiveness. For example, the attachment between preposition ل (pronounced as LAM, equivalent to the English letter L and means to or for) and nouns is simply neglected in light 10 and thus, a word like لدرجة (meaning: to the degree) will be preserved in light 10 although the attached preposition should be eliminated so as to be grouped with the original word درجة. Another example for such a prefix that is not included in light 10 is the preposition ب (equivalent to the English letter B), e.g. باسم (meaning: in the name), which will not be stemmed using light 10. Both letters are also considered in Arabic as clitics. Accordingly, it is not always possible for light 10 to deal with the clitics problem and consequently with proper nouns or grammatical nouns that are attached to them, as well.

Examples for antefixes and prefixes that are also not included in light 10 include the conjunction between the letter و (the letter WAW) and the two preposition ل and ب as in وللدماء and وبالدماء (meaning: for the bloods and with the bloods, respectively). Examples also include the conjunction between the letter ف (FAA) and the letter ب (BAA) to form فب as in فبالوطن (meaning: By the motherland). The prefixes ولل and فل are also not included in light 10 although they are often attached to nouns.

Second, when further analysis is done on light 10, it is noticed that Arabic verbs are partially ignored in the listed prefixes and suffixes. Consider the verbs تتجادل and جادل (meanings respectively: arguing and argued). Using light 10, verbs will not be stemmed to the same group as there is no prefix to be removed from their beginnings and as truncation in light 10 is focused on grammatical and proper nouns. Another example for the ignorance of verbs in light 10 is the prefix فل, which is usually used to emphasize doing the action (verb in this case). For example, in a word like فليكتب (meaning: you should write), which consists of the verb كتب and the prefix فل, light 10 stemmer will maintain the word as it appears during indexing and thus, other inflectional verbs, e.g. يكتب, كتبا, will not be grouped with فليكتب. It should be noted that the prefix فل can be also used with nouns. Thus, one of the major aims of the proposed Extended-Light is to consider some of the neglected prefixes and suffixes of both verbs and nouns in light-10. Note that the same arguments also apply for suffixes, rather than prefixes, that occur with both verbs and nouns. For instance, the absent pronouns in Arabic like كم and هم (meanings, respectively: your, their) are not included in

light 10 and thus a word like قاتلوكم or بيتكم (meaning: "they fight you" and "your house") will be indexed and stemmed separately from the words قاتل and بيت, for examples, although both words have the same semantics to their original words.

For these reasons and performing a deep analysis, our proposed Extended-Light stemmer adds more prefixes and suffixes, beside those described by light 10, so as to account for nouns as well as verbs. The new added prefixes are ل, ب, تت, فب, وب, ول, فل, ولل, وبال and the new extended suffixes are ت, هم, نا, هما, تي, وا. Table 1 illustrates the final sets of the strippable prefixes and suffixes in the proposed Extended-10 stemmer.

As it can be noticed in Table 1, the number of the prefixes in the proposed Extended-Light stemmer is greater than their peers in light 10. In the proposed stemmer prefixes are definite articles, conjunctions (prepositions, clitics and clitics attached to prepositions) or some letters that are often added to verbs, such as تت, according to additive verb rules in Arabic. It is also noticed that the number of the suffixes in the proposed stemmer adds additional elements to the suffix set that were introduced by light 10. For instance, suffixes in the Extended-Light stemmer includes the letter ت (pronounced as TAA), which is used in Arabic for singular first person masculine, second person masculine and third person feminine, as in لعبت (meanings can be: I played, you played and she played). The suffix set also include both the first and third person pronouns that indicate the number and gender of both verbs and nouns like نا, as in أكلنا (meaning: we ate) and هما, as in بيتهما (meaning: their house for dual masculine).

One might ask if we remove the prefixes like ب and ل from the beginning of a word, for example, as the Extended-Light stemmer proposes, how we could handle words like وليد (meaning: the proper noun Waleed) or لقمان , which is another proper noun. Note that light 10 stemmer stems these nouns as ليد and لقم, respectively and thus, it results in mis-stemming as both nouns become un-understood words. An explanation to this phenomenon in light 10 is that the stemmer avoids removal of letters like ل since many proper nouns begin with this letter. So the question is how our proposed stemmer handles such a situation? The answer for this question falls in the use of the proposed heuristic rules, which remove only certain prefixes and suffixes under certain conditions.

The underlying assumption behind our proposed algorithm is that since the majority of Arabic words are derived from tri-literal or quad-literal roots then any resulted stem for a word should not be less than 3 letters and should not exceed 4 letters, too but under certain criteria. This is totally different from the assumption behind light 10, which stated that stemmed words may be consisting of 2 or 3 letters under certain conditions. For instance, a word like وجه (meaning:

Table 1. The List of prefixes and suffixes in Extended-Light.

Removing from front (prefix)	Removing from end (suffix)
ال, وال, بال, كال, فال, لل, وبال, ولل, فل, ول, وب, فب, تت, و, ب, ل	ها, ان, ات, ون, ين, يه, ية, هـ, ة, ي, وا, تي, هما, نا, هم, ت

face) will be stemmed to وج, which is meaningless word consisting of 2 letters, when using light 10 stemmer, whereas a word like لقمان will be stemmed to لقم, which is a chaotic word for the original proper noun that consists of three letters. The problem, however, may even be much worse, when such a word is erroneously grouped with the verb لقم (meaning: to stuff). Examples include, but are not limited to, words like صحون (meaning: the plates), ساعة (meaning: a clock or an hour) and السودان (meaning: the Sudan). Using light 10 stemmer, the three words will be to ساع, صح and سود, respectively due to the stated rule which allows words to be consisting of 2 or 3 letters. To avoid such problems the proposed stemmer changes the rule while considering words characteristics. Our stemmer attempts to maintain words consisting of three letters and four letters as they appear, e.g. وجه and السودان in the examples above will be stemmed as وجه and سودان, in which only the definite article ال was removed.

Bearing in mind this discussion, the proposed algorithm contains the following three steps:

Step 1: in step one, the conjunctions و (pronounced as WAW), ب (equivalent to the English letter B) and ل (equivalent to the English letter L) are removed if and only if the remainder of the word is greater than 3. For examples, words وجد (meaning: found or passion) and بسم (meaning: in the name) will be preserved when step one is applied as both the letters و and ب are preserved too, and thus, they will not be handled as conjunctions, but as parts of the two words instead. On the other hand, for a word like لساعة (meaning: for one hour), the letter ل will be eliminated as the number of the remaining letters, after removing the letter ل, is greater than 3. On the algorithm of Extended-Light, the idea is also extended to include verbs. In light 10 stemmer a verb like تتنافسون (meaning: they are to compete for something) will be stemmed to تتنافس, which would result in an under-stemming problem as the new stem will not be clustered with the original root نافس. Thus, our proposed stemmer attempts to mitigate such types of problems.

Step 2: in the second step, the algorithm truncates the prefixes. This is can be achieved by firstly matching the word with the prefixes listed in the defined set. If any matched prefix is encountered, the algorithm removes that prefix from the input word if and only if the retained stem contains 3 letters or more; otherwise, the algorithm didn't eliminate the prefix.

Step 3: in step 3, the algorithm focuses on suffixes. As in step 2, it matched the suffixes first starting from right to left. When there is a part of the word under stemming matches a suffix, the algorithm removes that suffix. Before removing the suffix the algorithm checks the length of the possible stem. If the length is fewer than 4 letters, then it leaves the entire term; otherwise, the algorithm returns the stemmed term.

Table 2 shows some examples for Arabic words that were stemmed with both the proposed Extended-Light and light 10 stemmers. The next subsection describes the second technique that has been also proposed in this paper for stemming Arabic words.

Table 2. Some examples for Arabic words stemmed by the proposed Extended-Light and light 10 stemmers.

The Word	Meaning in English	Extended-Light	Light 10
الساعة	the clock	ساعة	ساع
أعلنت	I announced	اعلن	اعلنت
شركة	the company	شركة	شرك
للضمان	for the guarantee	ضمان	ضم
بالتالي	the next	تالي	تال
لدرجة	to the degree	درجة	درج
أعمالهم	their works	اعمال	اعمالهم
البطون	the bellies	بطون	بط
ليوم	for a day	يوم	ليوم

3.2. Linguistic-Based Stemmer

In this paper, we also developed another technique for stemming Arabic texts. The technique is based on some Arabic morphological rules and syntactic knowledge. But, it should be noted that the proposed Extended-Light stemmer, which was described above, can be implemented by its own or with this second proposed linguistic stemmer.

The premise made in the proposed linguistic stemmer is that since the nature of the Arabic morphological system is complex, it is believed that the process of stemming should not be dependent on a specific approach, light stemming only for example. Instead, our hypothesis is that good Arabic stemmer should allow different ad-hoc scenarios—depending on what type of word is to be stemmed—for stemming Arabic inflected words. To achieve this goal, the proposed linguistic stemmer is a combined approach that considers the analysis level of the words that are to be stemmed with the proposed Extended-Light stemmer. This would help in shaping which stemming approach is to be used.

From that perspective, the proposed stemmer is made up of some clues/sub-components, each of which has a certain role to accomplish in the process. **Figure 1** plots the major steps of the solution. At first, each word is matched against some predefined lists of noun and verbal patterns. The set of patterns employed for nouns is different from the one that is used for verbs. In the next step, words that are valid to be nouns or verbs (has the same pattern in both lists) are automatically tagged using an Arabic POS tagger. At the end of this step, words are clustered into two different classes: verbs and nouns. For nouns, the developed light stemmer, Extended-Light, will be used. If the word is classified as a verb then a root-based stemmer, particularly Khoja, will be employed. The complete details of the proposed linguistic solution are provided below.

3.2.1. Classifying Verbs from Nouns

In order to be able to achieve the goal of using different stemming mechanisms,

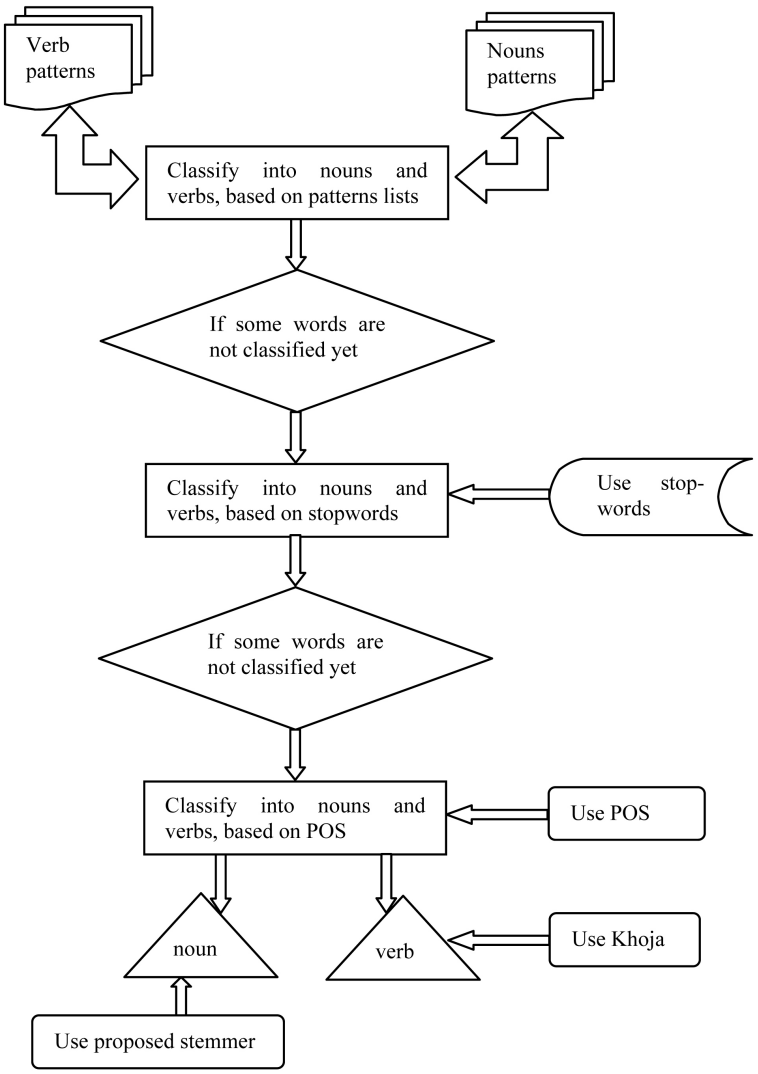

Figure 1. The steps of the proposed lingistic based stemmer.

verbs should be identified from nouns. This could be done by making use of some morphological rules and syntactic knowledge of Arabic. The foundation made here is that some patterns are valid only for verbs, while others are only valid for nouns. Thus, prior to applying the first step in the proposed solution, every word is appropriately rhymed to its pattern. The main rhym in Arabic as it was previously illustrated is the pattern فعل (f-à-l), in which the pattern preserves "f", "à" and "l" in the same order. By making use of this strategy, the corresponding pattern of every word is obtained.

During the same step, every word is examined, after it has been patterned, against two predefined lists of patterns: one for verbs and the other for nouns. These two key set of patterns are different as the set of patterns that are used for nouns in Arabic are not similar to the used ones for verbs. The two lists were compiled from several grammatical Arabic sources and have been revised by some Arabic grammarian experts. Patterns for verbs include, but are not limited

فعلل, فوعل, فيعل, فنعل, فعلى, فعلن فعل, يفعل, أفعل, تفعل, فاعل, تفاعل, استفعل, افتعل, انفعل, to افعال, افعول, افعوعل... etc. All of the patterns are derived from the Arabic syntactic rules, e.g. additive rules for verbs. Examples for nouns patterns are مفعل, أفاعيل, مفعول, مفاعيل, مفعلة... etc.

Thus, to accomplish the task of classification into verbs and nouns, the two sets of patterns are used. This technique is different from the one that has been used by Ababneh, *et al.* [22] in two points. First, in that study only single list of patterns for all words were used. Second, there was an invalid assumption in that study concerning some rules for words that would be preserved without stemming as they matched some certain patterns. For example, a word like كامل (meaning: the proper noun Kamil), which matches the pattern فاعل, will be preserved in Ababneh's study as it matches the entry فاعل in the stated list. This assumption is not true as there are many verbs that have the same pattern like قاتل (meaning: he fought). In addition to these different points between the proposed linguistic stemmer and the study of Ababneh [22], there is another major difference that is the use of POS in the proposed linguistic stemmer, as it will be illustrated later in this subsection.

During the same phase, the preceding words to the word under processing, are employed also to identify verbs from nouns with a hypothesis that some words, especially those are imperfect verbs (like كان, ما زال) and stop words precede nouns, e.g. إلى, كان, إن, من, بين, whereas others precede verbs, e.g. لن, لما, لم, عندما. The principle here is based on some grammatical rules in Arabic which says that that some words precede nouns while others precede verbs.

One major advantage for the use of only predefined lists of patterns for verbs and nouns and the use of the preceded words to identify verbs from nouns is that the process relatively distributes the burden of the identification phase as it minimizes that burden and changes it to only simple check against a predefined lists of some patterns (in the case of the predefined lists of patterns") and on some frequent imperfect verbs and stopwords (in the case of using only the preceded words). On the same time, the processes suppress the need for a POS tagger since it minimizes the additional overload performance required to identify the word clusters and reducing any undesired performance penalty that may occur due to the use of POS taggers.

Thus, using both the predefined lists of verbs and patterns and the preceded words, words under stemming are classified. However, an ambiguity may occur during this process. This happens when a word does not match any of the rhymed patterns or it has entries in both noun and verb pattern lists. For instance, both the words سالم (meaning: the proper noun Salim) and قاتل (meaning: he fought) are rhymed to the pattern فاعل and thus, if the traversed word is one of them, then an ambiguity problem will appear.

If an ambiguity or uncertainty about the word under processing occurs, an Arabic POS is used. In this paper, the Stanford Arabic POS Tagger has been used [33]. Stanford Arabic POS tagger is developed by the Stanford group of natural

language processing and it is a maximum entropy POS tagger [33]. The tagger is able to use both following and preceding tag contexts and it supports the idea of broad use of lexical features. The tagger also considers the features for disambiguating of tense forms of verbs and the features of disambiguating particles from prepositions and adverbs. As it is claimed by its developers, the tagger has the ability to tag the majority of the Arabic words fed to it.

Thus, for any ambiguous word, its containing paragraph only is fed to the tagger. This is a good feature accredited to the proposed linguistic stemmer, as the use of the POS tagger only when needed will not result in slowing the retrieval process. Thus, to avoid any slower performance for the tagger while on the same time attempting to preserve the context in which the word occurs, only the paragraph in which the word occurs is fed to the tagger. As the paragraph in which the word appears is fed to the tagger, the latter produces its POS. Since many tags can be generated by the tagger, an application code has been written to cluster the different tags into only noun or verb. For examples, tags like NN (produced by the tagger for nouns), DTNN (for a definite article attached with a noun) and PTNNS (for plural nouns that are attached to a definite article) are all collapse a single tag noun, while the different categories of verbs like VB (for the surface form of verbs) and VBG (for present verbs) are classified into a single tag called verb. Thus, by making use of the two lists of patterns, preceding words and the POS, words are tagged as nouns or verbs.

3.2.2. Producing Stems

After words are marked with nouns or verbs, words are then stemmed. Stemming for nouns in the proposed linguistic stemmer is different from stemming for verbs. On one hand, it was shown that the good feature of heavy stemming techniques that are based on morphological analysis like Khoja, is that they maintain POS distinctions [34] and since they retrieve all the related text, they reduce the index size significantly. Thus, if Khoja stemmer is employed for the purpose of stemming only verbs, then this will minimize over-stemming problem, which solely related to light stemming techniques. Accordingly, if a word is tagged as a verb, then it will be stemmed using Khoja, which is a root-based stemmer.

On the other hand, it was shown that light-stemming techniques are robust as they preserve the meanings of words. This feature of light stemming approaches is important for Arabic nouns, which represent a tremendous part of Arabic words. Thus, if a word is tagged as a noun in the proposed linguistic stemmer, the proposed Extended-Light stemmer will be used for indexing that word. This would results in minimizing the under-stemming problem, which is a major drawback for light-stemming techniques. The rationale behind this minimization has twofold. On one hand, the use of the proposed Extended-Light stemmer would result in reducing the impact of the under-stemming problem because the stemmer has been extended to include more prefixes and suffixes that were not

covered by light 10. In addition, the algorithm itself has been modified. On the other hand, since only nouns (not every word as in light 10) will be stemmed by the proposed Extended-Light stemmer, the effect of the under-stemming difficulty will be reduced also as the problem is originated from stemming verbs to different clusters.

4. Experimental Setup

To evaluate the proposed stemmers, TREC 2001 Arabic corpus has been used. It contains 383,872 documents compiled from Agence France Presse (AFP) Arabic Newswire during the time period of 1994 to 2000. The collection contains also 25 Arabic topics with equivalent versions in English and French. Besides the 25 topics, additional 50 topics from TREC 2002 were used in the experiments, resulting in 75 topics. Relevance judgments for the query set are also provided in both TREC 2001 and TREC 2002. In the experiments, the Arabic topics were used. In particular, Arabic titles with their description were used as queries.

Prior to indexing, texts were firstly normalized. At first, diacritical marks (like ً) were removed. The kasheeda (known also as tatweel), which is an Arabic stylistic elongation used for cosmetic writing as in عـــــادل instead of عادل, was also eliminated. Punctuation marks were also removed after they were used in tokenizing the texts. Following this, a letter normalization process to unify orthographical forms of letters was also executed. Due to orthographic variations for some characters in Arabic, the process of letter normalization often renders some different forms of some letters with a single Unicode representation. The letter normalization that had been performed includes:

- Replacing the letters ALIF HAMZA (أ،إ) and ALIF MADDA (آ) with bare ALIF (ا);
- Altering the final un-dotted YAA (ى) with dotted YAA (ي);
- Replacing the final TAA MARBOOTA (ة) with HAA (ه); and
- Modifying the sequence ىء with ئ.

Texts in documents had been tokenized on white space and punctuation marks. All experiments were conducted using the Lucene IR System that uses the Okapi IBM BM25 weighting. Lucene is an experimental information retrieval system that has being extensively used in previous editions of the CLEF, NTCIR and TREC joint evaluation experiments. The Apache Software Foundation describes Lucene as a high-performance search engine with many full-featured libraries to process and manipulate texts. Before populating texts in the Lucence with the appropriate stemmed terms, words were tagged in the experiment of the linguistic stemmer, while they are not in both light 10 and Extended-Light stemmers runs. Stopwords were also eliminated after they have been used in the proposed linguistic stemmer. The used stopword list was the one included in the Lucence. The average precision was used to measure retrieval performance and the statistical Student's t-test measure was used to compare significance of differences among the conducted experiments.

Three official runs were conducted. The first run, which was called Light 10, makes use of the light 10 stemmer and represents as a baseline, to which other two experiments would be compared. Light 10 is a widely reported baseline in Arabic IR studies. The second experiment tested the proposed Extended-Light stemmer alone as it was described earlier in the paper. This experiment run was called ExtendedS. The third experiment, which was called LingStem, is conducted to show the impact of using the proposed linguistic stemmer, in which nouns are stemmed in a way different to verbs. As described earlier, the proposed Extended-Light stemmer is being included in this experiment.

5. Results

Table 3 shows the average precision obtained for the three runs, while Figure 2 shows the comparison of the three curves of the average precision at 11 recall points of the 75 queries for the three experiments.

As shown in the figure, both the proposed stemmers (ExtendedS and LingStem) are consistently better than light 10. The difference is even evident at the majority of the precision-recall points. In particular, light 10 performed worse than the

Table 3. Compares the Average Precision for the three runs (light 10, Extended-Light and linguistic based stemmers).

	Light 10 Light-10 Baseline	ExtendedS Extended-Light	LingStem Linguistic Stemmer
Average precision	0.351	0.369	0.397
Percentage improvement (over light-10 Baseline)	-	5%	13%

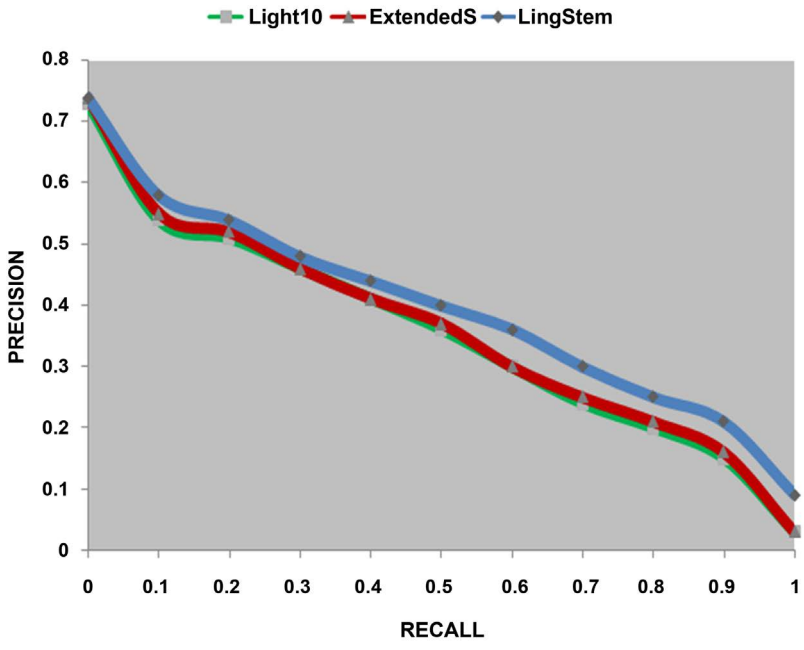

Figure 2. The three curves of the average precision at 11 recall points of the 75 queries.

proposed stemmers. This relatively worst performance was caused by the fact that light 10 does clustering words with the same meaning (those are semantically related to each others) to different conflation classes, although the language, meaning Arabic, conflates many words from a single stem or a single verb.

In terms of average precision, the proposed Extended-light stemmer (ExtendedS run) yields 5.13% in the retrieval performance over light 10 (with 0.369 average precision) and it performed significantly better than baseline. The difference is statistically significant (p-value < 0.05). An explanation for this notice is that affixes that were added to the proposed Extended-Light stemmer have a real impact on the stemming process. This is especially true, if we consider the affixes that were added in the proposed stemmer to account for verbs. Such affixes make the stemmer able to group variety of verbs and/or words into the same conflation class, unlike light 10, which always suffers from the under-stemming problem.

Consequently, the latter problem affects performance solely as it reduces the possibility of matching between posted queries and index documents. Nevertheless, it was expected that the difference in retrieval between the two stemmers (light 10 and Extended-Light) would be larger than what occurs. A possible explanation for this phenomenon is that the majority of the Arabic words affixes are already included in light 10, but yet, there is still a difference between retrieval performances of the two stemmers.

Comparing the proposed Extended-Light stemmer (ExtendedS run) to developed linguistic stemmer (LingStem run), which includes the use of the Extended-Light stemmer itself, the latter achieved 0.397 average precision and it yields 13.1% and 7.56% improvement in retrieval performance over light 10 and the proposed Extended-Light stemmer, respectively. The observed differences are statistically significant (p-value < 0.05). This improvement for the proposed linguistic stemmer stems from the fact that the major problem of light stemming techniques in general is the under-stemming, in which words with a single meaning are stemmed to different clusters. With regards to stemming verbs in light stemming approaches in general and in our experiments in particular, the under-stemming problem is widely spread due to two major reasons. Firstly, the absence of verbs' affixes in light-stemming approaches. As it was described earlier, light-stemming techniques focus on truncating nouns affixes, rather than verbs' affixes. Secondly, there is a relatively large number of rules to derive more verbs form each single tri-literal verb, for example, as in the verb قطع (meaning: cut), which can be conflated to قاطع, تقاطع, اقتطع, استقطع, تقطع, انقطع, يقطع... etc. As a result for this phenomenon, both light 10 and the proposed Extended-Light stemmer fail to group many verbs that have a single meaning into the same cluster, while the proposed linguistic stemmer does. Recall that the latter stemmer performs a classification step into verbs and nouns firstly.

On the other hand, the use of the proposed Extended-Light stemmer, which is

included in the linguistic stemmer, for nouns only contributes also in improving the performance of the LingStem run. Since additional suffixes, especially those related to nouns, and extra clitics have been included in the proposed Extended-Light stemmer, and since the latter stemmer is being used in the linguistic stemmer for nouns only, the performance stemming technique for nouns in LingStem becomes much better. Consider for examples the suffixes هما and هم. Both suffixes were not included in light 10 as thus words like بيتهما and بيتهم (meanings: their house for both dual and plural masculine, respectively) will not be grouped together or with the word بيت. But, the proposed linguistic stemmer considers such difficulties and attempts to overcome this complicated morphological structure of the Arabic language by allowing different stemming techniques according to a well pre-created classification step.

6. Conclusion and Future Work

The work presented above shows the importance of stemming to highly morphological languages such as Arabic. However, since the language is rich, the employed stemming technique could have a significant impact on improving retrieval performance of Arabic texts.

In the paper, two different stemmers were proposed. The first stemmer is a developed version for the best and most known stemmer for Arabic, which is light 10. Results showed that the chosen lists of affixes that are to be removed from Arabic words plus the heuristic rules that are often used in the truncation process of a certain stemmer could have a significant impact on its efficiency. Since the affixes to be tripped off in the proposed light stemmer have been chosen carefully and the heuristic rules of truncation have been well controlled, the developed light stemmer outperformed light 10 and difference is statistically significant. We believe that the success of light stemming approaches in general is caused by that the majority of Arabic words are nouns. But, this does not mean that verbs' affixes should be ignored. Accordingly, the proposed light stemmer adds extra affixes (some clitics) and modifies the truncation rules so as to consider both verbs and nouns.

The reported results also showed that the superior stemming technique could be achieved by using more than one approach for stemming. This is what the proposed linguistic stemmer, which is another developed stemmer in this paper, reported in the presented results as it achieved the best improvement on retrieval over both light 10 and the proposed stemmer. Using morphological analysis for classifying words into POS tags could be employed for determining which technique is to be used. However, in such a task, it is important to set up the environment carefully so as to avoid any performance load that can be produced by the POS taggers. What our proposed is what linguistic stemmer does. Results showed also that using different stemming techniques is the best solution for Arabic. Clustering words into different classes (*i.e.* nouns and verbs) and using different approaches could have a real effect on retrieval performance. Extract-

ing syntactic knowledge of preceding words (especially those imperfect verbs, prepositions and stop words) of the words under stemming and/or rhyming those words according to some list-driven patterns, could minimize the need for POS tagger, which always results in making the retrieval process slow.

In the future, the focus of the work will be on extracting more morphological rules as the language is very rich in its derivational system and including those rules in further study. This could have an impact on reducing the need of using a POS tagger. There is also an eye in the future works on the impact of using the proposed stemmers with query expansion techniques.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Miniwatts Marketing Group (2018) Internet World Stats Usage and Population Statistics. http://www.internetworldstats.com/stats7.htm

[2] Simpson, A.K. (2009) The Origin and Development of Nonconcatenative Morphology. Ph.D. Thesis, University of California, Berkeley.

[3] Mustafa, M., Eldeen, A.S., Bani-Ahmad, S. and Elfaki, A.O. (2017) A Comparative Survey on Arabic Stemming: Approaches and Challenges. *Intelligent Information Management*, **2017**, 39-67.

[4] Saad, M.K. and Ashour, W. (2010) OSAC: Open Source Arabic Corpora. The 6th International Conference on Electrical and Computer Systems (EECS'10), Lefke, 25-26 November 2010, ,118-123,

[5] Mustafa, M. (2013) Mixed-Language Arabic-English Information Retrieval. Ph.D. Thesis. University of Cape Town, Cape Town.

[6] Manzour, I. (2018) Lisan Al-Arab. http://www.lesanarab.com/

[7] Hegazi, N. and El-sharkawi, A. (1985) An Approach to a Computerized Lexical Analyzer for Natural Arabic Text. *Proceedings of the Arabic Language Conference*, Kuwait, 14-16.

[8] Mustafa, M. and Suleman, H. (2011) Building a Multilingual and Mixed Arabic-English Collection. *The Proceedings of the* 3*rd Arabic Language Technology International Conference* (*ALTIC*), Alexandria, 9-11 October 2001.

[9] Khoja, S. and Garside, R. (1999) Stemming Arabic Text. Computing Department, Lancaster University, Lancaster.

[10] Buckwalter, T. (2002) Buckwalter Arabic Morphological Analyzer Version 1.0. Linguistic Data Consortium, University of Pennsylvania, Philadelphia.

[11] Darwish, K. (2002) Building a Shallow Arabic Morphological Analyzer in One Day. *Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages*, Philadelphia, 11 July 2002, 1-8. https://doi.org/10.3115/1118637.1118643

[12] Xu, J., Fraser, A. and Weischedel, R. (2001) TREC 2001 Cross-Lingual Retrieval at BBN. *TREC* 2001, Gaithersburg, 13 November 2011, 68-78.

[13] Al-Sughaiyer, I.A. and Al-Kharashi, I.A. (2006) Rule Parser for Arabic Stemmer. In: Sojka, P., Kopeček, I. and Pala, K., Eds., *Text, Speech and Dialogue*, TSD 2002. Lec-

ture Notes in Computer Science, Vol. 2448, Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-46154-X_2

[14] Al-Shalabi, R., Kanaan, G., Ghwanmeh, S. and Nour, F.M. (2007) Stemmer Algorithm for Arabic Words Based on Excessive Letter Locations. 4*th International Conference on Innovations in Information Technology* (*IIT*'07), Dubai, 18-20 November 2007, 456-460. https://doi.org/10.1109/IIT.2007.4430444

[15] Darwish, K. and Oard, D.W. (2003) CLIR Experiments at Maryland for TREC-2002: Evidence Combination for Arabic-English Retrieval. *TREC* 2003 *Proceedings*, College Park, February, 2003.

[16] Aljlayl, M. and Frieder, O. (2002) On Arabic Search: Improving the Retrieval Effectiveness via Light Stemming Approach. In: *Proceedings of the* 11*th ACM International Conference on Information and Knowledge Management*, ACM Press, New York, 340-347. https://doi.org/10.1145/584792.584848

[17] Kadri, Y. and Nie, J.Y. (2006) Effective Stemming for Arabic Information Retrieval. *Proceedings of the Challenge of Arabic for NLP/MT Conference*, Londres, 3 October 2006, 68-74.

[18] Chen, A. and Gey, F. (2002) Building an Arabic Stemmer for Information Retrieval. In: *TREC*, NIST, Gaithersburg, 631-639.

[19] Nwesri, A.F.A., Tahaghoghi, S.M.M. and Scholer, F. (2005) Stemming Arabic Conjunctions and Prepositions. *Lecture Notes in Computer Science*, **3772**, 206-217. https://doi.org/10.1007/11575832_23

[20] Larkey, L., Ballesteros, L. and Connell, M. (2007) Light Stemming for Arabic Information Retrieval. In: *Arabic Computational Morphology*, Springer, Berlin, 221-243.

[21] Diab, M., Hacioglu, K. and Jurafsky, D. (2004) Automatic Tagging of Arabic Text: From Raw Text to Base Phrase Chunks. In: *Proceedings of HLT-NAACL: Short Papers*, Association for Computational Linguistics, 149-152. https://doi.org/10.3115/1613984.1614022

[22] Ababneh, M., Al-Shalabi, R., Kanaan, G. and Al-Nobani, A. (2012) Building an Effective Rule-Based Light Stemmer for Arabic Language to Improve Search Effectiveness. *The International Arab Journal of Information Technology*, **9**, 368-372.

[23] Larkey, L.S., Ballesteros, L. and Connell, M.E. (2002) Improving Stemming for Arabic Information Retrieval: Light Stemming and Co-Occurrence Analysis. *Annual ACM Conference on Research and Development in Information Retrieval: Proceedings of the* 25*th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Tampere, 11-15 August 2002, 275-282. https://doi.org/10.1145/564376.564425

[24] Xu, J. and Croft, W.B. (1998) Corpus-Based Stemming Using Co-Occurrence of Word Variants. *ACM Transactions on Information Systems*, **16**, 61-81. https://doi.org/10.1145/267954.267957

[25] Mustafa, S.H. and Al-Radaideh, Q.A. (2004) Using N-Grams for Arabic Text Searching. *Journal of the American Society for Information Science and Technology*, **55**, 1002-1007. https://doi.org/10.1002/asi.20051

[26] Xu, J., Fraser, A. and Weischedel, R. (2002) Empirical Studies in Strategies for Arabic Retrieval. *Annual ACM Conference on Research and Development in Information Retrieval: Proceedings of the* 25*th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Tampere, 11-15 August 2002, 269-274. https://doi.org/10.1145/564376.564424

[27] Hmeidi, I.I., Al-Shalabi, R.F., Al-Taani, A.T., Najadat, H. and Al-Hazaimeh, S.A.

(2010) A Novel Approach to the Extraction of Roots from Arabic Words Using Bi-grams. *Journal of American Society for Information Science and Technology*, **61**, 583-591.

[28] Al-shammari, E.T. and Lin, J. (2008) Towards an Error-Free Arabic Stemming. In: *Proceedings of the* 2*nd ACM workshop on Improving Non English Web Searching*, ACM, New York, 9-16. https://doi.org/10.1145/1460027.1460030

[29] Mansour, N., Haraty, R.A., Daher, W. and Houri, M. (2008) An Auto-Indexing Method for Arabic Text. *Information Processing and Management*, **44**, 1538-1545. https://doi.org/10.1016/j.ipm.2007.12.007

[30] Boubas, A., Lulu, L., Belkhouche, B. and Harous, S. (2011) GENESTEM: A Novel Approach for an Arabic Stemmer Using Genetic Algorithms. *International Conference on Innovations in Information Technology*, Abu Dhabi, 25-27 April 2011, 77-82. https://doi.org/10.1109/INNOVATIONS.2011.5893872

[31] Al-Serhan, H. and Ayesh, A. (2006) A Triliteral Word Roots Extraction Using Neural Network for Arabic. *International Conference on Computer Engineering and Systems*, Cairo, 5-7 November 2006, 436-440. https://doi.org/10.1109/ICCES.2006.320487

[32] Eldesouki, M., Arafa, A. and Darwish, K. (2009) Stemming Techniques of Arabic Language: Comparative Study from the Information Retrieval Perspective. *The Egyptian Computer Journal*, **36**, 30-49.

[33] The Stanford Natural Language Processing Group (2008) Arabic Natural Language Processing. https://nlp.stanford.edu/projects/arabic.shtml

[34] Levow, G.A., Oard, D.W. and Resnik, P. (2005) Dictionary-Based Techniques for Cross-Language Information Retrieval. *Information Processing and Management*, **41**, 523-547. https://doi.org/10.1016/j.ipm.2004.06.012