Journal of King Saud University – Computer and Information Sciences 29 (2017) 164–170

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

# Enhancing Arabic stemming process using resources and benchmarking tools

Younes Jaafar [a,*], Driss Namly [a], Karim Bouzoubaa [a], Abdellah Yousfi [b]

[a] Mohammadia School of Engineers, Mohammed Vth University – Rabat, Morocco
[b] FSJES, Mohammed Vth University – Rabat, Morocco

ARTICLE INFO

*Article history:*
Received 16 April 2016
Revised 4 November 2016
Accepted 21 November 2016
Available online 2 December 2016

*Keywords:*
Arabic stemming
Evaluation
Benchmark
Evaluation corpus

ABSTRACT

Many approaches and solutions have been proposed for developing Arabic light stemmers. These stemmers are often used in the context of application-oriented projects, especially when it comes to developing information retrieval (IR) systems. However, Arabic light stemming, as the process of stripping off a set of prefixes and/or suffixes, is a blinded task suffering from problems such as incorrect removal, vocalization ambiguity, single solution, etc. Moreover, each researcher claims that his/her stemmer reached a level of strength and accuracy quite high. However, in most cases, these stemmers are black boxes and it is not possible to access neither their source codes to verify their validity, nor the evaluation corpora that were used to claim such accuracy. Since these stemmers are very important for researchers, their comparison and evaluation is then essential to facilitate the choice of the stemmer to use in a given project. In this paper, we propose a new Arabic stemmer that gives solutions to the above mentioned drawbacks. In addition, we propose an automatic approach for the evaluation and comparison of Arabic stemmers that takes into account metrics related to the accuracy of results as well as the execution time of stemmers.



## 1. Introduction

Stemmers are basic tools used for many tasks that require text preprocessing, such as text categorization systems, text summarization systems, information extraction systems, etc. The stemming process includes the identification and the removal of affixes from a derived or inflected words and reducing them to their stems/roots. Different stemming approaches have been proposed for many languages including English, French, Turkish, and Chinese. Concerning the Arabic Language, there are two main stemming approaches (Otair, 2013): the root-based approach and the light stemming approach. Arabic is one of the Semitic languages, that differs from English, French, German, etc. Therefore, some Arabic stemmers reduce Arabic words to their roots instead of their stems (Al-Kabi and Al-Mustafa, 2006). In this article, we propose a third stemming approach that uses deeper validation of stems using lexicon resources.

The stemming process is important for researchers since it brings together words based on their lexico-semantic similarity. For example the words: "كتب" (he wrote), "كتبوا" (they wrote), "سيكتب" (he will write), "أكتبتم" (have you written?) have the same lexico-semantic content as "كتب" (he wrote) which leads to "the concept of writing". Thus, instead of dealing with four words, Arabic Natural Language Processing (ANLP) systems can handle one single word after reducing the list of words to the same stem. Therefore, queries or documents in IR systems can be represented using stems or roots rather than using the full original words. This operation reduces enormously the size of indexes of IR systems, which leads to a gain of space storage and processing time.

However, Arabic light stemming as the process of stripping off a set of prefixes and/or suffixes, is a "blinded" task suffering from problems such as:

- Incorrect removal: words starting with a string similar to a prefix, or ending with a string similar to a suffix will be truncated by mistake. For example, the analysis of the word "والده" ("his

* Corresponding author.
*E-mail addresses:* jayounes@yahoo.fr (Y. Jaafar), namly_driss@yahoo.fr (D. Namly), karim.bouzoubaa@emi.ac.ma (K. Bouzoubaa), yousfi240ma@yahoo.fr (A. Yousfi).

Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

http://dx.doi.org/10.1016/j.jksuci.2016.11.010
1319-1578/

father") with the Light10 stemmer (Larkey et al., 2007) gives the stem "د" considering "وال" as a prefix and "ه" as a suffix, whereas removing "وال" is an incorrect choice because it is a part of the stem.

- Vocalization ambiguity: removal of diacritic marks from the stemming output could lead to an ambiguous meaning of words. For example, analysis of the word "فكتابه" gives the stem "كتاب" considering "ف" as a prefix and "ه" as a suffix, but the stem "كتاب" allows several alternatives such as "كُتّاب" (school or authors), "كتَاب" (book), ... etc.
- Single solution: most of the available Arabic stemmers provide one solution in the stemming output, but according to Arabic language morphology, a word admits one or more different stems. For instance, the word "لهم" (for them) must return: the verb "لَهَمَ" (greed), the noun "لَهِمَ" (glutton), the verb "هَمَ" (interested) and the empty stem for the combination of the prefix "لَ" (la) and the suffix "هُمْ" (they).

Moreover, it is important for researchers to make an optimal choice when choosing a stemmer in the context of a larger project. To help researchers making this choice, it is essential to propose tools and approaches to evaluate and compare Arabic stemmers. The literature shows that researchers classify metrics for evaluating stemmers into two categories: (1) metrics related to the "strength" which describe changes made to words in order to produce stems, i.e. stronger stemmers are intended to make more changes to words to produce stems by removing characters, (2) and those related to the accuracy which describe how much these stems are correct. However, the Arabic content in the digital world has become so large that is difficult to neglect execution time in running text processing software. To our knowledge, there is no research that takes into account execution time for evaluating Arabic stemmers.

Therefore, our objective in this paper is twofold:

- Propose a new Arabic stemmer: SAFAR-Stemmer. This new stemmer is a stem-based one with a stem validation process using a lexical resource. SAFAR-Stemmer gives answers to the above cited drawbacks through the "multi-solution" concept. By offering multiple possible stems, it resolves the three aforementioned troubles. First, to correct the "Incorrect removal" deficiency, SAFAR-Stemmer comes up with a stems collection including all possible alternatives. Secondly, to satisfy the "Vocalization ambiguity" SAFAR-Stemmer provides a diacritized output. Thirdly, it offers several possible solutions according to the stemmed word composition in compliance with the morphological particularities guiding affixes agglutination, which resolves the "Single solution" problem. It should be noted that in stemming coherent texts, a word can always be assigned a unique stem, as the context provides the clues needed for disambiguation. However, there are many other cases where researchers need to stem words out of their contexts. That's why we believe that an Arabic stemmer should return all possible stems for a given word. Let us also mention that both of these two aspects (multiple solutions and vocalization) are not taken into consideration while evaluating stemmers in this article. This is because all other stemmers do not provide this information and it will be not fair to perform a benchmark in this case. That is to say, in this special case of evaluation, there is no added value if a stemmer returns one or multiple solutions. The evaluation is performed based only on the common form of output of all stemmers.
- Present a new reusable and generic solution to evaluate and compare Arabic stemmers. This is achieved using an evaluation corpus dedicated specifically to this purpose. We propose also a new metric of evaluation that combines metrics related to the accuracy of stemmers as well as their execution time. This new metric will allow researchers to make the optimal choice even if the metrics returned by stemmers are disproportionate. To give a concrete example of our evaluation, we selected three light stemmers namely: Light10 (Larkey et al., 2007), Motaz stemmer (Saad and Ashour, 2010), Tashaphyne (Zerrouki, 2016) in order to be compared with our new stemmer (SAFAR-Stemmer). It should be noted that our benchmarking solution can also handle root-based stemmers' benchmark.

The rest of this paper is organized as follows. The next section presents some stemming approaches and algorithms. In Section 3, we present our approach for the new Arabic stemmer. In Section 4, we present some works that deal with evaluating and benchmarking Arabic stemmers. We present also the evaluation corpus and some common metrics. Then we present our new metric for evaluating stemmers. Experiments and results are presented in Section 5. Finally, we present the conclusion and future works in Section 6.

## 2. Related works

In this article, we focus on Arabic light stemmers rather than root-based ones. Indeed, researches have shown that light stemmers give better results comparing to root-based approaches (Larkey et al., 2002). Therefore, it would be more appropriate to focus on more promising approaches.

Several Arabic light stemmer approaches and algorithms have been already proposed. They consist of removing the most common affixes from words and producing stems. Below are some examples of Arabic light stemmers.

Larkey et al. (2007) proposed several Arabic light stemmers and assessed their effectiveness for information retrieval using standard TREC data. The light stemmer, Light10, outperformed the other approaches. It has been widely used in Arabic information retrieval (Larkey et al., 2007).

Aljlayl and Frieder (2002) studied the stemming impact in improving Arabic information retrieval systems. For this, they have proposed two stemmers: a root algorithm based on the work of Khoja and a light stemming (LS) algorithm. Authors affirm that the LS algorithm significantly outperforms the root algorithm in IR. However, they do not provide evaluations for the two stemmers themselves.

Chen and Gey (2002) proposed also two Arabic stemmers for information retrieval: a Machine Translation (MT) based stemmer and a light stemmer. The test shows that the light stemmer performed better than the MT based stemmer in IR, but no evaluations were made to compare the two stemmers in terms of accuracy of their stemming results.

Rogati et al. (2003) presents an unsupervised learning approach for building an Arabic light stemmer. Authors compare results of their stemmer to GOLD which is a proprietary Arabic stemmer built using rules, affix lists and human annotated text. They claim their approach results in 87.5% agreement with GOLD.

Saad and Ashour (2010) proposed a light Arabic stemming algorithm to address the impact of text preprocessing on Arabic text classification. The system was integrated into WEKA (Hall et al., 2009) and RapidMiner (Hofmann and Klinkenberg, 2013) platforms.

We have selected three light stemmers: Light10, Motaz stemmer and Tashaphyne in order to compare their results with our stemmer and give a concrete example of use of our benchmarking system. It should be noted that we have focused in this article only on stemmers and not on morphological analyzers (benchmarking Arabic Morphological Analyzers has been done elsewhere (Jaafar