INTERNATIONAL JOURNAL OF AVANCED STUDIES IN COMPUTER SCIENCE AND ENGINEERING
IJASCSE VOLUME 5 ISSUE 11 2016

11/30/2016

Technique of Regular Expression for Arabic Light Stemmer

Omar Aldabbas
Department of Computer Science
Albalqa Applied University
Alsalt P.O. Box: 19117, Jordan
o.aldabbas@bau.edu.jo,

Riyadh Alshalabi
Department of Computer Science
Amman Arab University
Amman, P.O. Box: 2234, 11953, Jordan
shalabi@aau.edu.jo

Ghassan Kanaan
Department of Computer Science
Amman Arab University
Amman, P.O. Box: 2234, 11953, Jordan
Gkanaan@aau.edu.jo

Mohammed A.Shehab
Department of Computer Science
Jordan University for Science and Technology
P.O.Box 3030, Irbid 22110, Jordan
mohammed_Shihab@daad-alumni.de

Motasim Albdarnah
Department of Computer Science
Jordan University for Science and Technology
P.O.Box 3030, Irbid 22110, Jordan
Mhalbdarneh12@cit.just.edu.jo

Nizar Mahyoub
Department of Computer Science
Jordan University for Science and Technology
P.O.Box 3030, Irbid 22110, Jordan
naahmad12@cit.just.edu.jo

***Abstract*—Arabic light stemmer removes affixes from any word as well as stop words. It is considered as a text pre-processing task for many Natural Language Processing (NLP) applications such as text categorization, information retrieval , opinion mining , etc.. Many Arabic light stemmers were presented are depend on several techniques like the grammar-based, patterns-based , and mathematical rules- based. In this paper, a new Arabic stemmer based on regular expressions is proposed ,where it relies on regular expression to check out if the inputted word related to its text pattern or not. This stemmer is designed with two modes: (i) using only the proposed regular expression methodology , while (ii) hiring the Microsoft Word dictionary in addition to the proposed stemmer. The proposed methods achieved remarkable results that vary between 73.3% and 79.6%accuracy.**

Keywords: NLP; Light stemming; Arabic text; preprocessing; Regular expression

I. INTRODUCTION

The growth of local and international Arabic content is becoming larger and larger which leads us to come up with new mechanisms to process this large content and extract valuable information from it. The field of Natural Language Processing (NLP) is the field that concerned with the automated processing of human natural languages. It has many applications such as question answering ,machine translation , spilling correction, text categorization , and stemming.

Stemming is the process of truncating different morphologies of a word to result in a single morphem or base. As a result, text dimensionality is decreased and helps efficient computation. There is a difference between root or heavy stemming and light stemming. The former returns the words to their roots besides affix removal while the last only concerns removing suffixes, prefixes and stop words. Consequently, root stemming becomes more difficult to process and construct specially with Arabic words since there are so many words that do not depend on specific regulations[1]

Some errors may happen in the stemming process such as under stemming which occurs when two words derived from the same root are stemmed into two different words. Another error is called over stemming, which occurs when words derived from different roots are stemmed to the same Stem/Root [2].Many stemmers exist based on different

approaches like the grammar of the language itself, patterns in the language such as removing the prefixes, suffixes and stop words [3], or some mathematical rules [4].

Arabic language is considered one of the most difficult languages in the world because of its rough structure and complexity. Accordingly, returning words into their roots and stems becomes a big issue in Arabic Language studies. The importance of Arabic language arises because of the revolution of web technology, therefore and depending on some statistics, about 65.4 million of internet users are Arabs[5].

In this paper a new light stemmer based on regular expressions is introduced. Regular expression is a sequence of characters that forms a search pattern mainly for use in pattern matching with strings, or string matching. This mechanism is considered remarkable, especially when we talk about simplicity and run time progress . The use of regular expression can be very helpful in stemming by checking the similarity between the word and its pattern [6].

We build this system from the beginning using all the mentioned benefits about regular expression method, and in order to assess the constructed stemmer we entered 1000 Arabic words as a dataset to the tool. The results are encouraging and yield an even better accuracy when we used the Microsoft Word dictionary.

The rest of the paper is structured as follows: Section 1 gives a broad description about number of studies and related works in light steaming. Section 2 explains in details the methodology and its subsections present all the steps for constructing the system. Then section 3 presents the experiments and results showing the two setup modes (i.e. with and without word dictionary). After that in section 4,the errors analysis and results discussion are presented. Finally, section5 summarizes the whole study and provides guidelines for future work .

II. Related Work

Since root based stemming affects the retrieval process of the queried documents in most information retrieval applications, the need for light stemming arises. The reason behind not using the root based method is the ambiguity resulted from adding extra terms to the issued query. [1] Proposed a ruled based light stemming technique that improved the performance of the retrieving procedure . In their work, they introduced an algorithm to make light stemming on the entered Arabic word and can be summarized in the following steps: 1) Eliminate diacritic such as (fatha'a, dhamma, shaddah) from the word. 2) Make normalization on word by returning " إ,أ" to"ا"and"ه"to"ة"forexample.3) Eliminate the prefix "و" if the length of the word greater than 3.4) Remove definite articles such as "ال" from the beginning of the word if found. 5) Eliminate any extra suffixes that will make the length of the word greater than or equal 3. 6) If the length of the word still exceeds 3, remove the extra prefixes. They found from their experimental results in the TREC data set that any stemmed words returned to less than 3 letters will be considered fault, since all Arabic roots are trilateral or quad-literal words. They made a comparison between the original, light, and root stemming techniques after applying the AIRE information retrieval system , consequently they found that their technique outperform the others in term of enhancing the performance of the retrieval process .

Aljlayl and Frieder in 2002 enhanced the retrieval results on IR systems by presenting their light stemmer (light10). The ir stemmer depends on morphological analysis, and hence considered as the most open source light stemmer vastly used. Their work was expanded from their previous implement SIGIR 2002 by introducing different queries that gathered from TREC 2001 and TREC 2002 conducted with two morphological analyzers to assess their current stemmer . By driving the same steps by Ababneh and et al. mentioned in the previous passage, the resultant stemmer (light10) gave accuracy that outperforms the other stemmers presented in TREC 2001 and 2002. But in return, they couldn't achieve a high performed stemmer based on morphological analysis in compare is on with the others such as indexing roots and indexing stem or phological analyzers . There a son of that comes as the in sufficiency of Arabic morphological analyzers to perform better than the simple stemmers, and some factors lead to this degradation such as :the propagated mistakes of names, misunderstanding of the entered queries since IR systems understand queries in terms of collection of words instead of bags of unigram, the robustness of light stemmers that can't accept every word in the single sentence, and finally the disability of IR systems to understand the correct level of confusion [7].

Larkey and Margaretin 2007 proposed five algorithms for Arabic light stemming as enhancement of the TREC- 2002 light stemmer. Their algorithms based on two main methods to achieve the improvements: The first one is by maintaining the exclusion process of the suffix and prefix groups from the terms in the TREC-2002 stemmer so that a new affixation set can be added. And the second approach is by modifying the execution and the process of TREC-2002 algorithm in addition to the five proposed algorithms The ir implementation was being categorized into two sets: 1) the Suffix – Prefix method, taking into account the

algorithms that will remove the suffix letters recursively while dealing with the prefix lettersnon-recursively.2) The Suffix-Prefix-Suffix method, focusing on algorithms that will remove the first largest suffix letters, after that dealing with the first largest prefix letters elimination, then the second remaining largest suffix if found will be deleted as well The modification of TREC- 2002 is made by adding Suffix-Prefix-Suffix into the pre-defined algorithm. The ir approaches showed outstanding results among the TREC-2002 algorithm by stemming 1450 Arabic words and give meaningful results in a power of 20-30% more than that in the TREC-2002 technique [8].

Leah and et. al. in 2002 proposed four main techniques about Arabic stemmer. In their work, queries were normalized by changing some characters such as:"آ" ,"إ" ,"أ" to "ى" ,"ا" to "ي", and "ة" to "ه". After that, they started no. of stemming operations: the first one is the light stemmer in which they deployed four stemmers : light , light2, light3 and light8, accordingly they found that the best one was light8, but it still has problems like some words in Arabic language containing vowel letters such as" و" as one of word characters. The second operation is morphology analysis that first removes all suffix and prefix characters from words, then check a list of patterns and return them to their roots. The third operation is Simple Stemming that just used to remove vowel characters ا ,و ,ي ,ء. Finally, the fourth one is Co-occurrence Analysis that uses the Khoja stemmer with EMIM (expected mutual information) by measuring the Co-occurrence of a word if it is expected . The results of this research were divided into three parts: without query expansion, with English query expansion and with English and Arabic query expansion. In the first part, light8-s stemmer got the best accuracy 0.379 average precision. While in the other two parts, light8-s got the best average precision[3].

Al-Omari and et. al in 2012 presented a new Arabic stemmer which is based on mathematical rules and some relations between letters instead of Arabic root patterns. The presented algorithm has four main steps which are:
1)Reading a word, then finding its number of letters. 2)Identifying apposition to start the algorithm steps using a mathematical formula, and then removing that letter with the two neigh boring letters. 3) Searching for the extracted word in a dictionary of Arabic roots. If it's found, then stop searching. If not, then move to the right by one letter to get a new word then start searching the dictionary again. The algorithm will keep doing this until either a root found or no more letters exists to the right. 4)If the root is not found go back to the position computed in the second step and shift to the left by one character and search for the extracted word in the dictionary until either a root found or not. ARBLS showed better performance compared with Khoja's stemmer and detected some flaws in it[4].
Elrajubi in 2013 proposed an improved Arabic light stemmer. This stemmer consists of eight steps which are:
1)Dividing text into words. 2)Searching in stop word list. 3) Normalization. 4) Searching in Irregular words list. 5)Removing letter "و" (and) .6) Removing the prefixes and suffixes from the words .7) Applying correcting word rules. 8) Checking and getting the result. This stemmer was compared with light 10 stemmer and showed a better accuracy rate of stemming words where light 10 accuracy was (66%) and the proposed stemmer accuracy was (88.25%)[9].

Affixes	WORDS				
Suffix	ابرزهم	ابرمها	ابعدها	اجتماعات	احتفظوا
Prefix	الاتحاد	الاتفاقية	بيبالغ	تدرس	كالاطعمة

III. METHODOLOGY

In this section we explain in details how we dedicated our effort to make this work more accurate and reliable. This section illustrates all phases conducted in this study including collecting the data set and building the tool.

A. Dataset Collection

The data set constructed from[9] and consists of 1000 words that are collected automatically by extracting them from articles and also manually an notated in-house by two linguists in Arabic language (the authors of this study). The words have several lingual categories and POS. Moreover, we ensured that each word is useful by including all the affixes and stop words needed in the experimentation to evaluate the proposed stemmer such as: suffixes (ون, ات, ين etc..), prefixes (بـ,سـ, الـ,….. etc.), as shown in Table I and stop words (ذلك,كان,في,…..etc.) as shown in Table II

Table I : Affixes Sample

Table II : Stop word Sample

B. *The Proposed Stemmer*

In this section, we explain in details how our stemmer is built and on which method it depends. Regular expression is a special language that is used to check if the existing text related to a particular pattern or not. The proposed light stemmer is built using C# programming language and it makes use of the mentioned technique in order to detect extra characters (affixes) in Arabic words. For example: as we know there are prefixes characters in Arabic words such as:{"الـ"}, this pattern is saved in text file outside stemmer code to help user to edit those characters in future. The regular expression in this stemmer loads this prefix with many other prefixes to create prefix pattern. So if the input has this pattern like "الـلاعب", then the regular expression matches this pattern "الـلاعب". After that the stemmer can remove this pattern after detected it by using remover function and also for suffixes : ("الـلاعبـون"). Many techniques to detect these characters exist, however; the ambiguity in morphology arose confusion during the light stemmer construction. The tool was built taking into consideration two preprocessing tasks:

1. *Affixes removal*

Unlike other languages, Arabic language is rich in its structure and morphology. Its words sometimes appear with prefixes and suffix characters that give the word a different POS or meaning in the sentence. Appendix 1 gives a list of such letters. Removing affixes will make the entered text sparser to ease and help with text analysis by reducing its dimensionality.

2. *Stop words removal*

Besides the affixes, the tool also eliminates the stop words. Those words have no meaning or influence on the processed text. Thus, many studies suggested that they must be removed in order to assist in the computation efficiency.

C. *The StemmingAlgorithm*

The proposed stemmer uses the three lists of affixes and stop words list (mentioned in Appendix A) to carry out the stemming process for Arabic words as follows:

- The first step is loading each list, then it uses these lists to build a particular function. Building a function is the main role that constructs a main pattern for the regular expression. So, for each list mentioned in the appendix, the pattern function and the main regular expression pattern are built accordingly.
- After loading and building main patterns for each list, the stemmer runs and removes any prefixes, suffixes, stop words, and articles such as (الـ) that means "The", by mapping each original word with the regular expression pattern and removing the excessive letters consequently. The light stemmer must capture the articles, or prefixes at first by checking if the articles pattern is matched, then it will remove it from the word and continue to check the existence of the prefix pattern. Finally, the stemmer will check if there is any suffix at the end of the word. Otherwise, if the tool does not match the defined articles, then it will check directly the prefix and suffix patterns consequently.

D. *Stemmer with Microsoft Word Dictionary*

In this work, we built two modes of a light stemmer: The first mode uses regular expression without using the Microsoft word dictionary. This mode causes many spelling errors. Those errors occurred due to removing the additional characters in a particular word, which could convert it to completely another different one. That is, some words changed in meaning and gave no sense. For example, the word "اتضح " that means "appear" converted to "ضح" which has no meaning in Arabic language after removing the prefix "ا".

Stemmed word	WORDS				
Stop word	في	لكن	كان	ذلك	لازال

Hence, we added a dictionary to check the part of speech (POS) of the stemmed word, but also this technique failed to resolve this problem. The reason of that is in some cases the word after stemming has the same POS but with different meaning. Therefore, we proposed a new idea to resolve this problem by using the Microsoft word dictionary to check the similarity between synonyms of the word before and after removing the excessive characters. If the word and its synonym are different after the stemming process, then the word meaning is changed. In each stemming process, the algorithm checks the similarity between the old and the new word. If there is any change in

the meaning, then it never removes these characters. Fig. 1 shows the proposed algorithm and all the necessary steps of the stemming process.

```
1  Load addition characters files
2  Build and Initialize regular expression patterns
3  IF word match defined articles pattern THEN
4  New_Word = regular
   expressionpatterns.Remove(word)
5  IF  Check_ Synonyms_ Similarity(New_word,word)
   is sameTHEN
6  Word=New_word
7  ENDIF
8  ELSE IF word match prefix character    pattern
   THEN
9  New_Word = regular
   expressionpatterns.Remove(word)
10 IF Check_Synonyms_ Similarity(New_word,word)is
   sameTHEN
11 Word=New_word
12 ENDIF
13  ENDIF
14  IF  word match suffix character patternTHEN
15 New_Word = regular
   expressionpatterns.Remove(word)
16  IF  Check_ Synonyms_
    Similarity(New_word,word) is sameTHEN
17 Word=New_word
18 ENDIF
19 ENDIF
```

IV. Experiments and Results

In this section, we applied the constructed stemmer on the collected data set which described in the last section. We entered the 1000 words to see how many words the tool can stem correctly (Trueresults) and how many words are stemmed in correctly (Falseresults) so that we can measure the accuracy of the proposed stemmer. This process was done in two phases:

- The first phase: we conducted experimentation without the word dictionary utilization, and we observed that 733 (out of 1000 words) were stemmed correctly. These outcomes are acceptable and very promising.
- The second phase: we applied the stemmer associated with The Microsoft Word dictionary - we expect to acquire remarkable results. Surprisingly, we found out that 794 (out of 1000 words) were correctly stemmed. In conclusion, adding the Microsoft word dictionary enhances the behavior of the proposed light stemmer.

4. Assessment Strategy

In literature, many papers assessed their work with respect to a recall and precision criteria. In this work we evaluate the performance of the stemmer according to two respectful assessment methods: Accuracy and Error rate.

Accuracy is defined as the fraction between the correctly stemmed words and the total annotated words [9].

$$Accuracy = \frac{true\ results}{all\ collection\ data} \qquad (1)$$

While, the error rate is defined as the ratio between the unmatched results with the whole annotated collection [9].

$$Error = \frac{false\ results}{all\ collection\ data} \qquad (2)$$

According to Eq. 1, the accuracy of our stemmer without the dictionary mode is ‘0.733’ i.e. 73.3% of the entered data is stemmed correctly while with the dictionary mode it yields ‘0.794’ i.e. 79.4%accuracy.

And the error rate, according to Eq. 2 without dictionary mode is ‘0.267’ i.e. 26.7% of the total words are incorrectly stemmed where as 0.206 i.e. 20.6% of the total words are unmatched with the annotated words. Figure 2 shows the accuracy and error rate of the different stemmer modes.

V. Analysis of Errors

Although the built light stemmer achieved very acceptable accuracy, still there are some lacks through its behavior and functionality. In this section we explain those cases and their causes, which we plan to overcome in the future. Most of the errors arise during the excessive letters removal phase, which changed the morphological structure and the grammatical group of the word.

In the first mode, by the absence of the word dictionary strategy we obtained:

- New words resulted from the stemming process such as after stemming the word "اتضح " which means "appear" into "ضح" that makes no sense in Arabic language.
- We thought that when we remove the "ات", which is the suffix of the plural feminine, and replace it with "ة", it will represent the feminine singular of the word. For instance, "راكبات" which means the plural feminine of the word "riders" stemmed to " راكبة" which is the singular of "riders", but unfortunately in some cases that doesn't work as we planned such in the case of "اجتماعات" that means "meetings" was turned to " اجتماعة"while the right word must be "اجتماع" which means "meeting".
- Some words are not Arabic and considered as scientific terms such as "ببتونات" that means "Peptones" or"تكنلوجيا" which means "Technology", hence the tool doesn't recognize the difference between them and the other Arabic words. So after the stemming process those words will be "بتونة" and "كنولوجي" which also have no meaning in Arabic.
- Words such "تحيات" which means "greetings" becomes " حية" which means "snake" after the stemming process, while the correct word must be " تحية" "greeting". There a son was that the tool doesn't recognize which characters are significant and part of the original word from those considered as affixes .

So we speculated that when we use the Microsoft dictionary, the tool might address most of the mentioned mistakes and produce correct out puts. It works in some cases such as "احتفظوا" which means "they kept" that in the former method gives us "حتفظ" whereas adding the dictionary results in "احتفظ" which is the correct form. But for some cases it fails to produce the right stemmed word such as "اشعاعات" which means "radiations" and gives us "اشعاعة" which is wrong but the dictionary finds it right. Accordingly, the addition of the dictionary (Dictionary mode) augmented the accuracy by approximately 6% over the traditional mode.

VI. Conclusion

This paper introduces a new light stemmer system based on regular expression technique. It is aimed to stem Arabic words by removing prefixes, suffixes and stop words. In this work, the stemmer is designed with two modes: (i) without using the dictionary and, (ii) with using the Microsoft Word dictionary. The results are satisfying where the stemming accuracy is 73.3% without using Microsoft Word dictionary whereas it increases by 6% with Microsoft Word dictionary to be 79.4%. The proposed light stemmer has some drawbacks in its functionality and the error percentage occurred because the stemmer cannot deal with some irregular Arabic words. As a future work, we intend to improve the accuracy of this system by adding more restrictions to avoid the mentioned errors. Also, we plan to use new techniques to detect exceptional cases for Arabic words.

REFERENCES

[1] M. Ababneh, R. Al-shalabi, G. Kanaan and A. Al-nobani, "Building an Effective Rule-BasedLight temmer for Arabic Language to Improve Search Effectiveness," International Arab Journal of Information Technology, vol. 9, no. 4, p. 5,2012.

[2] E. T. Al-Shammari and J. Lin, "Towards an error-free Arabic stemming," in Proceedings of the 2nd ACM workshop on Improving non english web searching, New York,2008.

[3] L. S. Larkey, L. Ballesteros and M. E. Connell, "Improving stemming for Arabic information retrieval: light stemming and co-occurrence analysis," in Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information , New York, 2002.

[4] A. Al-Omari, B. Abuata and M. Al-Kabi, "Building and Benchmarking New Heavy/Light," in the 4th International conference on Information and Communication systems (ICICS 2013), Irbid, 2012.

[5] "INTERNET WORLD USERS BY LANGUAGE, " Miniwatts Marketing Group, 2001. [Online]. Available : http://www.internetworldstats.com/stats7.htm. [Accessed2015].

[6] Regular expression," Wikipedia, 2015. [Online]. Available: https://en.wikipedia.org/wiki/Regular_expression.

[7] M. Aljlayl and O. Frieder, "On arabic search: improving the retrieval effectiveness via a light stemming approach," in Proceedings of the eleventh international conference on Information and knowledge management, 2002.

[8] L. S. Larkey, L. Ballesteros and M. E. Conn, "Light Stemming for Arabic InformationRetrieval," Arabic Computational Morphology, vol. 38, p. 22,2007.

[9] O. Elrajubi, "An improved Arabic light stemmer," in International Conference on Research and Innovation in Information Systems (ICRIIS), 2013, Kuala Lumpur,2013.

[10] Mohamad Ababneh, Riyad Al-Shalabi, Ghassan Kanaan, Alaa Al-Nobani **(2012).**Building an effective rule-based light stemmer for arabic language to improve search effectiveness. **The International Arab Journal of Information Technology. 9(**4): 368-372.

[11] Ismail Hmeidi, Riyad Al-Shalabi, Ahmad T. Al-Taani, Hassan Najadat, Shaker A. Al-Hazaimeh. **(2010),** A novel approach to the extraction of roots from Arabic words using bigrams. **JASIST**: 583~591

[12] Al-Shalabi R., Ghwanmeh S., Kanaan G., and Nour F. **(2009).** Effect of Excessive Letter Location in Arabic Lexical Items: a Stemmer Algorithm Approach. **Abhath Al-Yarmouk Journal: Basis Sci. & Eng., 18(1).**

[13] Ghwanmeh, S.H., Kanaan, G., Al-Shalabi, R., Rabab'ah, S. (2009). Enhanced Algorithm for Extracting the Root of Arabic Words. **In CGIV 388-391**.

[14] Al-Shalabi R., Kanaan G., Ghwanmeh S., and Nour, F. **(2007).** Stemmer Algorithm for Arabic Words Based on Excessive Letter Locations. **Proceedings of the 4th International Conference on Innovations in Information Technology.** Nov 18-20, 2007. Dubi, United Arab Emirates.

[15] Al-Shalabi, R., Kanaan, G., and Muaidi, H. **(2003)**. New Approach for Extracting Arabic Roots. **Proceeding of the International Arab Conference on Information Technology**. Alexandria, Egypt.

Apindex A