ISSN: 0067-2904
GIF: 0.851

Modified Light Stemming Algorithm for Arabic Language

Rafal Ali Sameer*
Department of Computer Science, College of Science, University of Baghdad, Baghdad, Iraq

Abstract

Stemming is a pre-processing step in Text mining applications as well as it is very important in most of the Information Retrieval systems. The goal of stemming is to reduce different grammatical forms of a word and sometimes derivationally related forms of a word to a common base (root or stem) form like reducing noun, adjective, verb, adverb etc. to its base form. The stem needs not to be identical to the morphological root of the word; it is usually sufficient that related words map to the same stem, even if this stem is not in itself a valid root. As in other languages; there is a need for an effective stemming algorithm for the indexing and retrieval of Arabic documents while the Arabic stemming algorithms are not widely available. The current algorithm will perform preprocessing operations then matches the result word to Arabic patterns to get the stem of the word. This paper proposed a modified light stemming algorithm for Arabic Languages. As shown from the results, the proposed algorithm is an efficient algorithm.

Keywords: Stemming, stop words, Light stemming algorithm.

الخوارزمية المعدلة لاستعادة الجذور في اللغة العربية

رفل علي سمير*
قسم علوم الحاسبات، كلية العلوم، جامعة بغداد، بغداد، العراق

الخلاصة

استعادة الجذر هي خطوة معالجة مسبقة في تطبيقات استخراج اصول الكلمات وكذلك تعد ذات اهمية كبيرة في معظم انظمة استرجاع المعلومات. الهدف من استعادة الجذر هو تقليل الصيغ النحوية المختلفة للكلمة واحيانا صيغ الاشتقاق للكلمة الى صيغة الاساس (جذر او اصل) الكلمة، مثل استرجاع الاسم، الصفة، الظرف، الفعل الى الاصل الذي جاءت منه. الجذر لايكون بالضرورة مطابق للجذر النحوي للكلمة، عادة يكفي ان الكلمات ذات الصلة تؤدي الى نفس الجذر حتى لو كان هذا الجذر ليس الجذر الصحيح. كما في بقية اللغات نحتاج الى خوارزمية فعالة لفهرسة واسترجاع النصوص باللغة العربية حيث ان خوارزميات استعادة الجذور باللغة العربية لاتتوفر بصورة واسعة. في هذه الخوارزمية سوف نحتاج الى معالجة مسبقة للكلمات العربية ثم نقارن الكلمة الناتجة مع مجموعة من صيغ الكلمات العربية لاسترجاع جذر الكلمة. في هذا البحث عدلت خوارزمية استعادة الجذور واثبتت النتائج كفاءة هذه الخوارزمية.

1. Introduction

Word stemming is an important feature supported by present day indexing and search systems. Indexing and searching are in turn part of Text Mining applications, Natural Language Processing (NLP) systems and Information Retrieval (IR) systems [1]. Stemming is defined as the conflation of all variations of specific words to a single form called the root or stem [2]. Stemming is usually done by removing any attached suffixes and prefixes (affixes) from indexed terms before the actual assignment of the term to the index. Text clustering, categorization and summarization also require this conversion as part of the pre-processing before actually applying any related algorithm [1].

*Email: rafalali@scbaghdad.edu.iq

Many Stemming algorithms have been developed for a wide range of languages including English, Malay, Latin, Indonesian, Swedish, Dutch, German and Italian, French, Slovene, and Turkish, Bangla, Chinese. For Arabic Language there are three different Stemming approaches: the root-based approach, the light stemmer approach, and the statistical stemmer approach yet no a complete stemmer for this language is available [3].

Building an effective stemming algorithm for Arabic language has been always a hot research topic in the IR field since Arabic language has a very different and difficult structure than other languages, the official language of 23 countries, and one of the official languages of the United Nations. Arabic is the most widely spoken language after Chinese.

Classical Arabic "الفصحى" is also formally called Modern Standard Arabic (MSA) is the formal language in the Arabic world for reading and writing, and is viewed as the only true version of the language by all Arabs, MSA is used to write all books, newspapers, magazines, and media text [4].

2. Previous Researches

The strengths of stemming algorithms are varied and stemming errors are produced for every stemmer [5].There is little number of algorithms for Arabic language stemming. Some of the papers related to the produced algorithm are:

1. "A Rule-Based Arabic Stemming Algorithm" [2], this research proposed algorithms reported are either general in nature, or lack in the morphological aspect of getting to the correct Arabic stems.
2. "Effective Retrieval Techniques for Arabic Text" [4], the approaches described in this thesis represent an important step towards realizing highly effective retrieval of Arabic text.
3. "GENESTEM: A novel approach for an Arabic stemmer using genetic algorithms" [6], study exhibits a novel Arabic stemming algorithm which uses genetic algorithms and verbs pattern matching. This algorithm is based mainly on machine learning system and Arabic morphological rules or patterns. They produced an Arabic morphological analyzer capable to generate the Arabic root for any stream of Arabic words.
4. "Arabic Light Stemmer (Ars)" [7], paper proposed an Arabic stemmer dedicated to different Arabic dialects. They describe in their study new rule-based algorithm to extract stems from textual Arabic Gulf dialect.
5. "A new and efficient stemming technique for Arabic Text Categorization" [8], Light stemming refers to removing some defined prefixes and suffixes from the word without trying to deal with infixes instead of extracting the original root or recognizing patterns, it is not dictionary driven, so it is not required to have an Arabic word after removing suffixes.

3. Stemming

Stems are roots combined with derivational morphemes (generally using patterns) that attach to a word at the beginning (prefix), the middle (infix), or the end (suffix).Stems are the basic form of a surface word that can be inflected using other morphemes. Surface forms of Arabic words comprise two or more morphemes: a root with a semantic meaning, and a pattern with syntactic information. For example, the word "دروس" is a stem comprises the root "درس" and the infix "و" [4].

Stemming alone is less applicable to process strong morphological language such as Arabic but it is a very essential technique for processing such language, which requires a further effort of morphological analysis. Morphological (Morphology means the internal structure of words) is relatively weak in English language in compare with other languages like Arabic language where the morphology of Arabic language is very strong, complex, and sophisticated (for example, many variants maybe given for a word). Arabic language needs robust stemming techniques in order to process its complex morphological structure, absolutely morphological techniques are required to eliminate affixes (suffixes and prefixes) from words according to their internal structure [5], while we need more techniques that will be explained in this algorithm.

In stemming the 'stem' is obtaining after applying a set of rules but without bothering about the part of speech (POS) or the context of the word occurrence.

In stemming, conversion of morphological forms of a word to its stem is done assuming each one is semantically related. All word variants should map to same form (stem), after the stemming has been completed there are two points to be considered:

- Morphological forms of a word are assumed to have the same base meaning and hence should be mapped to the same stem.
- Words that do not have the same meaning should be kept separate.

These two rules are good enough as long as the resultant stems are useful for our text mining or language processing applications [1].

4. Stemming Advantages and Stemming Errors

Stemming simplifies the searchers' job by making the IR system satisfy their information need. Stemming reduces the size of index terms and thus reduces the size of the index (inverted file), too.
As the size of index terms is reduced, the storage space and processing time are reduced as well.
By the use of stemmers, Words in the collection must be organized into groups, multiple errors are produced and may be used to compare and evaluate stemmers.
There are two measurements of errors in stemming algorithms called *understemming error* (If two words belong to the same class of development in meaning been changed to different origins, (i.e. more than possible expressions or terms removed)), and *overstemming error* (If two words belong to the same class of development in meaning changed to the same origin then the stemmer went on correct, (i.e. not much of the expressions or terms are removed)) [5].

5. Normalization

This process is usually conducted as a pre-processing step before stemming. The following three replacements steps affect the spelling ambiguity. Normalization which functions as follows:

• Replace إ، أ، آ with ا.
• Replace ى with ي.
• Replace ة with ه. [5].

The normalization process preserves the word sense intact. Similarly for words that contain"hamza-under-alif" such as (إنسان), which means "human" in English, can be written as (انسان). Similarly, the letter "ta-marbotah" (ة) that occurs at the end of the Arabic word which indicates mostly the feminine noun is, in most cases, written as "ha" (ه) which makes the word ambiguous. To resolve the ambiguity, we replace any occurrences of (ة) in the end of the word with (ه). For example, the word (حقيقة) alternately appears as (حقيقة) or (حقيقه) in Arabic text, and the sequence of "alif-maksoura" (ى) in L_{n-1} and "hamza" (ء) in L_n (ء ى) will be replaced to (ئ) "alif-maksoura-mahmozah". Similarly, (ء ي) replace the sequence of "ya" (ي) in L_{n-1} and (ء) in L_n to (ئ) [9].

6. Stop Words

Stop word lists drawn up for Arabic contain well-known pronouns, prepositions and function words. These lists differ substantially, and no single widely accepted list exists. Critically, most lists include a single version of each word, despite the fact that Arabic words have different forms. Despite this disagreement on the appropriate stopword list size and content, there is an agreement that removing them from Arabic text improves retrieval precision [4].

Stopwords have to be chosen carefully as they affect retrieval. In English for example, some queries might contain only stopwords, for instance, "to be or not to be". In Arabic, some function words can be spelt identically to proper nouns. The absence of diacritics makes it difficult to distinguish between such words unless we consider the context. For example, the word "على" could be (/alaa/(bove)) could be (/ali/ noun), and so on.

To use Arabic patterns, we modified the Khoja stemmer to check whether there is a match between a word and a list of patterns after stemming without further checking against the root dictionary [4].

7. Arabic Language Light Stemming Algorithm

Light stemmers remove a pre-prepared list of prefixes and suffixes, by comparing initial and ending letters of Arabic words with list of prefixes and suffixes and remove matching sequences that pass possible additional criteria, such as that the remaining string should contain at least three characters.

Characters are added at the beginning, the middle, or end of the root will be removed, but the base characters that match the pattern remain unchanged. After every prefix or suffix removal, the algorithm compares the remaining stem with the patterns [4].

8. The Proposed Approach Algorithms (Modified Light Stemming Algorithm for Arabic Languages)

This algorithm performs a match operation for a word against (68) patterns and returns three-letter root. These patterns are classified according to their length (4, 5, 6, 7, 8, or 9). It also uses a list of prefixes and suffixes that range in length from 1 to 3 characters.

1. The first process of this algorithm is checking the word if it is Arabic word (hasn't digits, hasn't foreign letter(s), the number of characters in word less than three characters), if one of these

conditions satisfy then ignore the word, unless all of these conditions doesn't satisfy then continue.

2. Normalizing the different forms of hamza "ؤ ،ئ ،ء", as well as normalizes the different forms of *alef* (آ،إ،أ) to *alef* (ا).
3. Check a word against (132) stopwords in (Table-1), if it is a stopword ignore word.
4. If the word is not a stopword the algorithm will check the word against list of prefixes, in (table-2) if it is matched remove prefix and return the word for further matches.
5. Compare the word against list of suffixes (Table-2) and remove them according to their relevant order (if there is more than one suffix).
6. At last the resulted word (word after processing) is to be matched with (68) patterns (table-3), word with three letters root match to the pattern "فعل". For example: the root "كتب" can be represented by the pattern"فعل" by mapping "كـ" to "فـ","تـ" to "عـ" and "ب" to "ل". a four letter word is compared with four character patterns; a fifth letter word is compared with fifth character patterns (and so on) to return three letters as stem (root) of the word. If no root is found, the original word is returned untouched.

For languages such as Arabic, it will be risk to make algorithm does not use dictionary to check the correctness of the resulted stem. Although, a lot of stemming algorithm use dictionary for comparing the output of stemming algorithm with correct Arabic word stored in database, this algorithm was depend on the output of the stemming algorithm only.

Table 1- Lists of Stopwords

No.	Stop Words	No.	StopWords	No.	StopWords
1	ابا	46	حول	91	لذلك
2	ابو	47	حيث	92	لعل
3	ابي	48	حين	93	لكن
4	احد	49	خلال	94	لم
5	اخا	50	دون	95	لماذا
6	اخر	51	ذا	96	لن
7	اخو	52	ذات	97	له
8	اخي	53	ذلك	98	لو
9	اذا	54	ذو	99	ليت
10	الا	55	ذي	100	ليس
11	الان	56	رغم	101	ما
12	التي	57	شيء	102	ماانفك
13	الذي	58	صار	103	مابرح
14	الذين	59	صبح	104	ماذا
15	اللتان	60	صبر	105	مازال
16	اللتين	61	ضحى	106	مافتئ
17	اللذان	62	ضد	107	مايزال

Continue to Table 1- list of stopword…

No.	Stop Words	No.	StopWords	No.	StopWords
18	اللذين	63	ضمن	108	متى
19	الي	64	ظل	109	مساء
20	اليوم	65	عل	110	مسي
21	اما	66	على	111	مع
22	امام	67	عن	112	مما
23	امس	68	عند	113	من
24	ان	69	عين	114	منذ
25	او	70	غير	115	نحو
26	اول	71	ف	116	نفس
27	اين	72	فقط	117	هؤلاء
28	اي	73	في	118	هذا
29	ب	74	فيما	119	هذه
30	بات	75	قبل	120	هل
31	بان	76	قد	121	هن
32	بد	77	ك	122	هنا
33	بدل	78	كان	123	هو
34	بعد	79	كذلك	124	هي
35	بعض	80	كل	125	هما
36	بل	81	كم	126	هم
37	بيت	82	كون	127	وسط
38	بين	83	كي	128	يكون
39	تحت	84	كيف	129	يلي
40	تكون	85	ل	130	يمكن
41	تلك	86	لا	131	يوم
42	ثم	87	لازال		
43	جدا	88	لاسيما		
44	حالي	89	لدي		
45	حتى	90	لايزال		

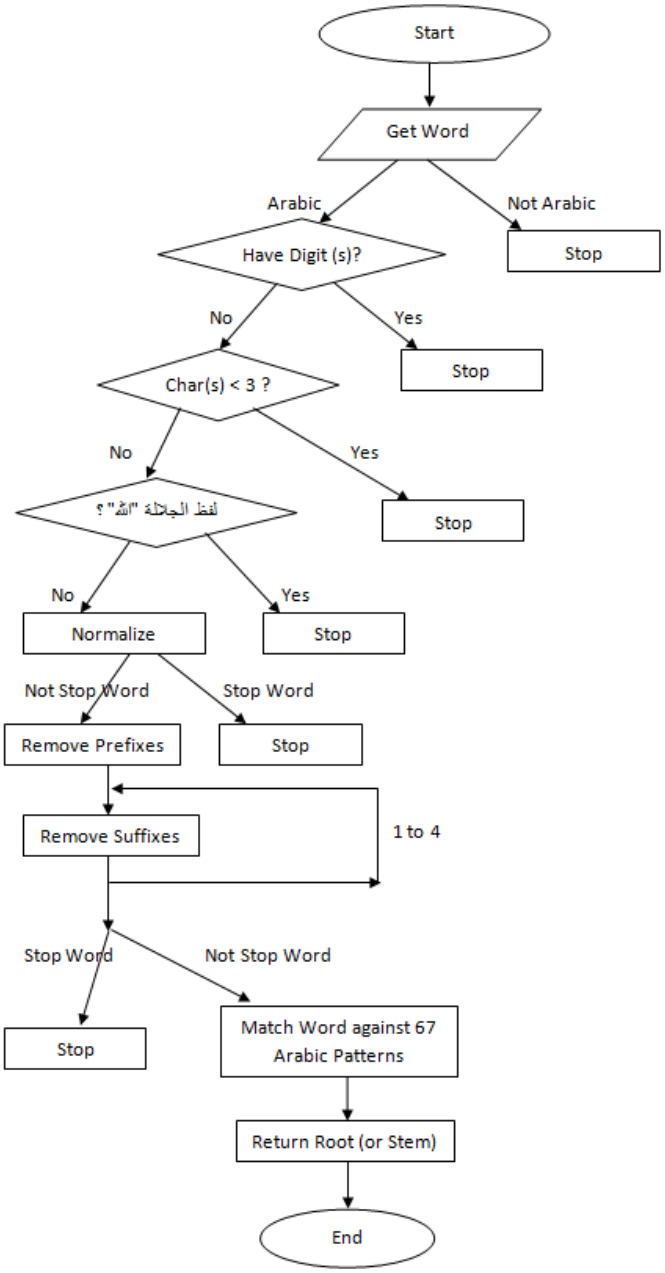
Table 2- List of Prefixes and Suffixes

No.	Proposed Suffixes	Proposed Prefixes
1	هما	بال
2	كما	فال
3	ات	كال
4	يه	ال
5	ته	لل
6	تي	وال
7	ان	و
8	ون	س
9	ين	ف
10	هم	
11	هن	
12	ها	
13	نا	
14	وا	
15	كم	
16	كن	
17	ني	
18	وني	
19	تم	
20	ه	
21	ي	

Table 3- List of Arabic Patterns

No.	Patterns	No.	Patterns	No.	Patterns
1	فاعل	24	تفعلة	47	فعالية
2	فعلى	25	تفعيل	48	مستفعل
3	فعلة	26	مفعال	49	مفاعلة
4	تفعل	27	مفعيل	50	مفاعيل
5	فعول	28	مفعول	51	منفعلة
6	فعيل	29	مفعلة	52	مفعيلة
7	فعال	30	مفاعل	53	مفعولة
8	مفعل	31	مفتعل	54	مفتعلة
9	أفعل	32	متفعل	55	متفعلة
10	فاعلة	33	منفعل	56	مفعللة
11	فاعول	34	مفعلل	57	تفاعيل
12	فعلاء	35	فعللة	58	استفعال
13	فعلان	36	فعللي	59	انفعالة
14	فعولة	37	فعلانة	60	إفتعالة
15	فعيلة	38	تفعيلة	61	افتعالي
16	فعلية	39	أفتعال	62	أفعالية
17	فعائل	40	انفعال	63	مستفعلة
18	فعالي	41	إفعالي	64	مفعولية
19	فعالة	42	إفعالا	65	متفاعلة
20	فواعل	43	أفعلاء	66	إستفعالي
21	فعائل	44	أفعلية	67	افتعالية
22	افعال	45	فاعولة	68	استفعالية
23	افعلة	46	فعلولة		

9. The Block Diagram for the Proposed Approach



10. Experimental Result

The words will be used for testing process are (10) different words, the resulted true stemmed word are (9), so the percentage for algorithm evaluation is:

$$\text{Result} = \frac{\sum WT}{\sum WC} \times 100$$

While,
WT: is the number of True tested word.
WC: is the number of tested word [10].
The percentage of correctness for the proposed algorithm for (14) words will be as follow:

$$\text{Result} = \frac{12}{14} \times 100,$$

Where the number 12 refers to true tested words and 14 refers to total number of tested words.
Result = 85 %

Table 4- Examples for Stemming using proposed approach

No.	Word	Actual Word	Resulting Root	Result Checking
1	تزخر	زخر	زخر	True (overstemming)
2	يجعلنا	جعل	جعل	True (overstemming)
3	الفسوق	فسق	فسق	True (overstemming)
4	ربه	ربب	ربه	Unchanged
5	فسيعملون	عمل	عمل	True (overstemming)
6	الواجب	وجب	وجب	True (overstemming)
7	استعمالاتها	عمل	عمل	True (overstemming)
8	رماهم	رمى	رما	Same spelling
9	تماثيل	مثل	مثل	True (overstemming)
10	اللذين	StopWord	اللذين	Unchanged
11	استقالة	اقال	سقل	False (understemming)
12	مخترعين	اخترع	خرع	False (understemming)
13	كالطير	طار	طير	Same spelling
14	سيقول	قال	قول	Same spelling

11. Conclusion

This approach is mainly dependent on the understanding of the Arabic morphology. Stemming approaches for European Languages such as English and French are not fully appropriate for the development of Arabic stemmers due to differences in the morphological structures peculiar to each of the languages as well as their semantic differences. This algorithm (Modified light stemming algorithm for Arabic Languages) is efficient and accurate for stemming the words that have different length and different affixes according to the experimental results but falsely stems proper names and foreign words.

References

1. Anjali Ganesh Jivani, **2011**. A Comparative Study of Stemming Algorithms, *International Journal Comp. Tech. Appl.*, India.
2. Tengku Mohd T. Sembok, Belal Mustafa Abu Ata, and Zainab Abu Bakar. **2011**. A Rule-Based Arabic Stemming Algorithm, Proceedings of the European Computing Conference.
3. Meryeme Hadni, Said Alaoui Ouatik, Abdelmonaime Lachkar, **2013**. Effective Arabic Stemmer Based Hybrid Approach for Arabic Texti Categorization, *International Journal of Data Mining & Knowledge Management Process*, Morocco.
4. Abdusalam F. Ahmed Nwesri, **2008**. Effective Retrieval Techniques for Arabic Text, School of Computer Science and Information Technology Science, Engineering, and Technology Portfolio, RMIT University, Thesis, Australia.
5. Mohammed A. Otair, **2013**, Comparative Analysis of Arabic Stemming Algorithms, *International Journal of Managing Information Technology (IJMIT)*, Jordan, 5(2).
6. 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, pp: 77–82.
7. Asma Al-Omari, Belal Abuata, **2014**. Arabic Light Stemmer (Ars), *Journal of Engineering Science and Technology*, School of Engineering, Taylor's University.
8. Hadni, M., Lachkar, A., and Ouatik, S. **2012.** A new and efficient stemming technique for Arabic Text Categorization, Multimedia Computing and Systems (ICMCS) International Conference, Morocco.
9. Mohammed Aljlayl, and Ophir Frieder. **2002**. On Arabic Search: Improving the Retrieval Effectiveness via a Light Stemming Approach, International Conference on Information and Knowledge Management, USA.
10. Marwan Ali.H. Ome, and Ma shi long. **2009**. Stemming Algorithm to Classify Arabic Documents, *Symposium on Progress in Information & Communication Technology*, China.