Improved Stemming Methods for Arabic Language for Enhanced Search Engine Efficiency

Ali Abrahem Alnaied
The Higher Institute of Science and Technology Tajoura
Information Technology Dept.
Tripoli , Libya
a_alnaied@yahoo.com

ABSTRACT

Year after year, many methods are being published to overcome the Arabic stem problem for successful retrieval of documents. Therefore, this research present a novel method to extracting Arabic stem. In our method we investigate Arabic morphology features. The main goal and advantage of our approach is to generate/extract stem by applying a set of rules and matches the relationship between some Arabic letters to find the root/stem of the respective words in order to uses as indexing term for the text searching in Arabic retrieval systems. Consequently, our method can be considered to operate around minimum morphological complexity, and also solve problems of conjunctions in Arabic such as prepositions and stopword that are linked directly to the word.

Indeed, these tasks are very hard and require an understanding of the meaning of a text and the ability to reason over relevant facts. Using only supporting facts. Thus, we have been tested our method using the EveTAR (2016) dataset on Arabic tweets and the obtained results show that our method results outperform the state-of-the-art results. Therefore, our method has been able to improve performance of Arabic stem and increases retrieval as

well as being active against any type of stem and we believe that it's difficult to develop new Arabic system retrieval method without uses a good morphology analysis support it.

Keywords: Natural language processing, Arabic information retrieval systems, Arabic morphology, Light Stemming algorithm, Arabic Stemming algorithms, Rule based stemming, Indexing, Word segmentation.

INTRODCTION

Every day, the internet offers huge volume of data to service users' needs, and many users on the internet needs to retrieve documents by using only a few words, or a query to fetch all information or documents that relevant to their search query. However, the quality of the search depends on the query's words; if query words were not precise enough, it may influence the search ability to retrieve the correct documents.

In fact, to ensure a satisfying level of precision for Arabic information retrieval systems it require decomposed words in the query into meaningful components before submitted to the retrieval system. So, in this study, we present new method which is powerful tool using to extract stem and morphology from word this process is depends on stem and morphology.

On the other hand, the advantageous of using stem and morphology in information retrieval systems is to conflate query terms into indexing terms. Hence, one of the most challenging morphologies in natural language processing is Arabic language. It can produce successfully user information needs in addition to saves time.

The essential component of a word is its stem; for instance, Arabic stems are different as compared to other languages like English. Arabic nouns can take the form of being plural, singular, dual, gender; feminine or masculine, and verbs can be present, past, future, and command verb. In contrast, stemming refers to a computational technique used to reduce words to their respective stems or roots. One disadvantage of existing Arabic stemmers is that they exhibit and are prone to immense stemming error-rates (Paice, 1994).

Therefore, in this work, as were mention early, a new method has been made for extract Arabic stem or root of word based on morphology features, which is a way below the acceptable level in query words of precision. This is because Arabic stem processing is still needing more research to offer any contribution that a larger framework such as information retrieval.

Related work

In recent decades, many works have been done in the field of developed Arabic themed information retrieval system, but there are still many weakness and problems facing the Arabic language most sufficient and reliable information retrieval systems immensely rely on morphological and stemming analysis but still a little deal with lemmatization. In this section, we will discuss what has been achieved on stemming and morphology in the literature and how stemming and morphological analysis impacts the retrieval of documents in Arabic.

Khoja's stemmer previously showed the first attempt to find the Arabic root by removal of prefixes and suffixes. Researcher (porter, 1980) developed the Porter tailored for the English

language. This stemmer leverages on two-step rewriting rules. This is achieved by removing approximately 60 different suffixes by (Al-Fedaghi & Al-Anzi, 1989). Up to now, the Porter stemmer has been documented to have an exemplary performance, especially in its precision and recall of evaluations. However, this stemmer has the drawback of being very aggressive in its creation of stems and ends up over stemming, and it was later used by several researchers.

(Larkey, et al., 2007) Show better retrieval efficiency, among described in light stemming; it merely removes prefixes and suffixes depends on a listed in a predefined. However, it does not guarantee the production of better results when evaluating experiments, as proposed by (Aitao & Fredric, 2002).

(Darwish, K, Abdelali, A., et, 2016) proposed FARASA a new method of Arabic segmenter. Which is more efficient in terms of the query time when compared to MADAMIRA. FARASA produces word segmentations; However, this stemmer technique cannot handle any infixes segmentations.

(M. M. Almanea, 2021) stem pattern rules, transitivity rules, and definiteness rules. After that, rules for extracting case-endings of imperfect verbs, noun phrases, nouns, proper nouns, adjectives, and adverbs were applied. The researchers used the LDC's Arabic Treebank to test the system, and the WER in case-endings was about 9.97%.

(Etaiwi, W., & Awajan, A, 2022) Most of graph-based Arabic NLP studies used a static graphs rather than dynamic ones, which could be explained by the complexity of dealing with Arabic language due to its structure and morphology.

In existence are numerous root extraction techniques for Arabic known as heavy stemming or stemming based root words, works by removing all affixes (prefix, infix, and suffix), and uses to improved Arabic information retrieval performance, shown by (Al-Shalabi, 1996).

(Arfath, et al., 2014) Is one of the best and most technique for Arabic and Arabic dialect processing tool designed for morphological analysis in a context that combines different aspects used systems for Arabic processing, they apply language and SVM and models in the predictions of word tags based on feature modelling component.

Information Retrieval and Arabic Language

Over the last decade, Arabic information retrieval has garnered significant attention due to increasing the Arabic text on the web. A considerable number of researchers share similar opinions on the benefits of morphology and stemming in Arabic information retrieval systems, especially for internet search engines; a problem exacerbated by the enormous amounts of data on the internet. Therefore, in this chapter we will emphasis more on important aspects of information retrieval systems like broken plurals, derivation, affixation, morphology, and language.

Arabic language is a Semitic language family, such as Aramaic and Hebrew, over 389 million people use it as their first language and around 140 million non-native speakers. Arabic spoken in a large area including of the Middle East, North Africa, most of the Arabian Peninsula, and other parts. Therefore, in linguistic word composed are different between Arabic and English, hence Arabic grammar is very different from English;

for example: Arabic offers more inflection word than English which are comes with the gender, numbers, person, noun, and adverb. While in English, inflection word can be coming with numbers within the sentence: (Office → Offices) and person within the sentence: (e.g., I play → he plays). In Arabic, there is no distinction is made between upper and lower case, and the rules for punctuation are much looser than in English. In addition to that, one big difference between the Arabic and English languages is that Arabic doesn't use abbreviations or capitalization, Arabic letters are only written in cursive, and Numbers are written from left-to-right.

Arabic is a semitic language, includes 28 alphabets with three short vowels, namely Alef (ا), Waw (و), and Ya' (ي). Words are written style in horizontal lines from right to left. The shape of each letter depends on its position in the word, the letter س (s) has different shape depends on its position in the word as shown in the following on Table 1.

Arabic Letter	Begins	Middles	Ends
س (s)	سـ	ـسـ	ـس

Table 1: Example of Arabic letters shape.

In addition, Arabic numerals are written from left to right, and most root contains of three letters, though there are also some roots whose contains more than three letters. And also, the prepositions and stop-words may come separately or linked directly with the same word, like ب, ف, ك, ل, و which means (in, so, as, for, and). Therefore, Arabic morphology can uses to reduce different grammatical word forms for nouns and verbs.

Our Method Stemmer Algorithm

This subsection discusses our method algorithm to find the stem or root of the word that uses as indexing term in the field of Arabic information retrieval systems. Our method algorithm works as follows:

Tokenization & Normalization

Arabic tokenization has been implemented in several solutions to resolve ambiguous words. For instance, characters can be written in different ways, such as character (ء) Hamza can be composed in different ways (أ, إ, آ). This cause more ambiguous as to whether the Hamza is present. Therefore, at most one token is assigned to each letter at any one time as follows:

- Replacing initial أ, آ, إ by ا
- Replacing final ى, ئ by ي.
- Replacing final ه by ة.

Keyword Extraction.

We represent our method steps to extract Keywords as follows:

1. Convert the user request text into words and put it into a list.
2. Check the lists whether prepositions or stop-word are found. If found, remove any matched from the list
3. Search our method Dictionary to find given terms in the list; if a match found, then extract root/stem if accepted on our method rules.
4. Else, if a match not found, do nothing.

Step1: Convert the user request text into words to create a word list by selecting the words that contain more than three letters.

Step 2: Check the created lists, if prepositions or stop-word found, if they found, then remove prepositions or stop-word from the list.

Step 3: Search in our method dictionary, if any match found in the given list, then extract root/stem based on our method rules. For example; if we give the word 'ولمدرس' (And for a teacher) to our method dictionary which is consist of three prefixes م (m), ل (for), and و (and). So based on our method rule 1, we will remove prefix ل (for) which refer to preposition, and prefix و (and) which refer to stop-word. So, we will get مدرس (teacher), which using as indexing term.

Step 4: if a match not found in our method dictionary, then not do anything.

More detail around how our method works as follows: it starts by receiving a request from user's query; then check if length of the three-letter word or more, and then seeks the query words in our method dictionary, if the word is found, thus then extract root or stem if accepted on our method rules. An example of this process: The word requested is مدرسة (school); when this word gives to other Arabic retrieval systems, it will return the stem مدرس (Teacher) by remove the suffix ة (taa), in this case the word meaning has changing. through the use our method rules No 2 as shown in the next Chapter, which says that if the word composed prefixes م (M) and suffixes ة (taa) joined together in the same word; thus, this case will a produce noun (always refer to places)., Hence, our method system suggested to keep the word formations as they are. So, our method system will extract the word مدرسة (school). This result is more precision, which we aims to have.

Experiments and results

In this work, we simply plan to verify effectiveness and the quality of our method performed with relevance judgments. So, we present an overview of the tests performed, as follow:

Dataset

Experiment was carried out with Arabic Test collections EveTAR on tweets that are comparable to similar Text Retrieval Evaluation Conference TREC. Test collections EveTAR are evaluation tools that are essential for advancing the state-of-the-art in the field of Arabic information retrieval that supports multiple information retrieval EveTAR includes a crawl of 355M which contained roughly 61946 articles on an Arabic tweet represented in Unicode and encoded in UTF-8, and covers 50 significant events for which about 62K tweets were judged with substantial average inter-annotator agreement.

Measures

In the literature, we have seen various methods to test the effectiveness of the Arab information retrieval system from relevant and irrelevant documents. Typically, evaluation measures are computed across multiple queries and averaged to produce a final score. Therefore, the primary evaluation measure used in this work is the mean average precision (MAP), in addition to the precision at 10 (P@10) and precision at 20 (P@20) in order to analyze the change in retrieval precision.

Mean Average Precision MAP which are defined as the following formula as following:

$$MAP = \frac{\sum \mathbf{n_{k,j}}\, AveP(q)}{\mathbf{Q}}$$

Results

In this section, retrieval performance of the proposed method our method has been compared using BM25 model and language model LM with Dirichlet. Furthermore, and the retrieved effectiveness was evaluated using MAP by using BM25 and LM model, in addition to, the precision at 10 (P@10) and precision at 20 (P@20) in order to analyze the change in retrieval precision. Therefore, Table 2 and Table 3 presents our experimental results, where the bold values denote the best results in each category. Thus, both Table 2 and Table 3 shown the results obtained for each system runs for 50 queries, these results are analizied in the next section.

	BM25 Model		
	MAP	**Prec@10**	**Prec@20**
Proposed Method	**0.34**	**0.63**	**0.59**
No stem	0.21	0.45	0.46

Table 2: Summary of the results obtained for MAP by using BM25 model.

	LM with Dirichlet smoothing Model		
	MAP	**Prec@10**	**Prec@20**
Proposed Method	**0.32**	**0.60**	**0.56**
No stem	0.18	0.29	0.28

Table 3: Summary of the results obtained by using LM with Dirichlet smoothing model.

Basically, the results presented in Table 2 and Table 3 clearly indicate that the proposed method is capable to solve successfully the research problems in high performance level, so the best retrieval performance for Arabic information retrieval systems was our proposed method.

Figures 1 shows precision and recall for the retrieval methods tested at p@10 points for proposed method. It shown that method approach has better performance than others.

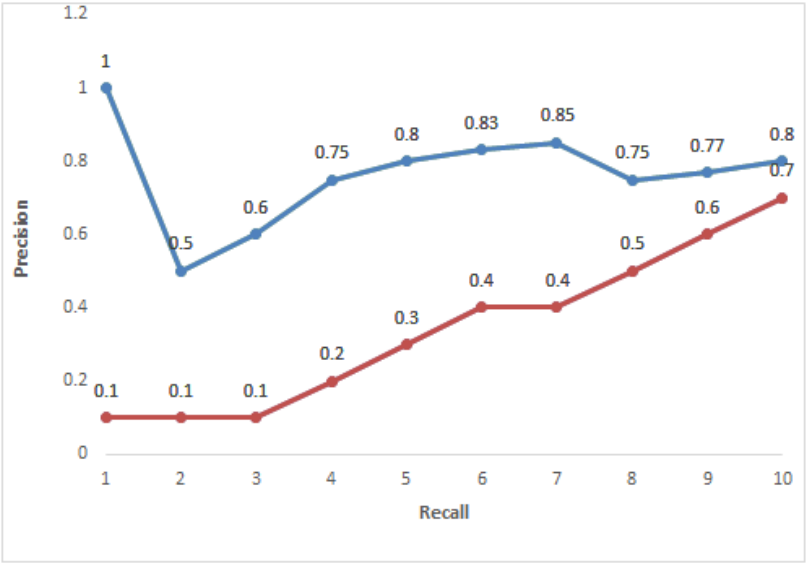

Figure 1: Precision-Recall achieved by using the proposed method

It is be noted that our method is highlighted by blue color.

Conclusion

Since Arabic is a highly inflected language, the most important research algorithms improved Arabic retrieval systems based on a morphology analyses and light stemming.

We have been investigated the effect of the morphological analysis (derivational and inflectional) on information retrieval performance. We found that our proposed method can be develops an light stemming, which is represented different words forms. And also, as a result, we found that word processing containing stem using our method is better than the light stemming as well as being strong against any type of stem.

References

[1]. Paice, C.D. An evaluation method for stemming algorithms. in SIGIR'94. 1994. Springer.

[2]. M. M. Almanea, "Automatic Methods and Neural Networks in Arabic Texts Diacritization: A Comprehensive Survey," IEEE Access, vol. 9, 2021.

[3]. Etaiwi, W., & Awajan, A. (2022). SemG-TS: Abstractive arabic text summarization usingsemantic graph embedding.

[4]. Al-Shalabi, R., et al. Stemmer algorithm for Arabic words based on excessive letter locations. in 2007 Innovations in Information Technologies (IIT). 2007. IEEE.

[5]. Darwish, K., H. Hassan, and O. Emam. Examining the effect of improved context sensitive morphology on Arabic information retrieval. in Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages. 2005. Association for Computational Linguistics.

[6]. Larkey, L.S., L. Ballesteros, and M.E. Connell, Light stemming for Arabic information retrieval, in Arabic computational morphology. 2007, Springer. p. 221-243.

[7]. Khoja, S. and R. Garside, Stemming arabic text. Lancaster, UK, Computing Department, Lancaster University, 1999.

[8]. Porter, M., An algorithm for suffix stripping. Program: electronic library & information systems. 1980.

[9]. Larkey, L.S., 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 retrieval. 2002. ACM.

[10]. Larkey, L.S. and M.E. Connell, Structured queries, language modeling, and relevance modeling in cross-language information retrieval. Information processing management 2005. 41(3): p. 457-473.

[11]. Khoja, S. APT: Arabic part-of-speech tagger. in Proceedings of the Student Workshop at NAACL. 2001.

[12]. Darwish, K, Abdelali, A., et al. Farasa: A fast and furious segmenter for arabic. in Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Demonstrations. 2016.