We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS BOOK CITATION INDEX INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Chapter

Applications of Mining Arabic Text: A Review

Qasem Al-Radaideh

Abstract

Since the appearance of text mining, the Arabic language gained some interest in applying several text mining tasks over a text written in the Arabic language. There are several challenges faced by the researchers. These tasks include Arabic text summarization, which is one of the challenging open areas for research in natural language processing (NLP) and text mining fields, Arabic text categorization, and Arabic sentiment analysis. This chapter reviews some of the past and current researches and trends in these areas and some future challenges that need to be tackled. It also presents some case studies for two of the reviewed approaches.

Keywords: text mining, Arabic language, Arabic text categorization, Arabic sentiment analysis, Arabic text summarization

1. Introduction

The massive increase in the amount and availability of online documents made retrieving and searching for information a difficult job and a problem for web users. The need for efficient and powerful tools to automatically categorize or summarize text has emerged. To overcome this problem, several methods and techniques have been proposed. One of these solutions is using different text mining tasks. Basically, text mining can be defined as the process of extracting knowledge from the massive amount of textual data. For this purpose, researchers in the fields of information retrieval, natural language processing, and data mining have investigated several types of text mining tasks and methods. These tasks include text categorization [1, 2] and text summarization [3].

2. Features of the Arabic language

The Arabic language is a popular language and is a member of the Semitic language family. It is the most widely spoken language of almost 330 million people and as a native language in more than 25 countries in an area spread from the Arabian Gulf in the East to the Atlantic Ocean in the West. Arabic is a highly structured and derivational language, where morphology plays a very important role [4].

The Arabic language ranks fifth among the top 30 languages spoken worldwide.

The Arabic language has three diversities, which include classical Arabic, modern standard Arabic (MSA), and colloquial Arabic. The classical Arabic is usually used for religious and historical scripts. The MSA can be found in today's written Arabic text in most formal channels. The colloquial or dialectical Arabic is

the spoken language in informal and social media channels. In addition, the dialects vary from one Arab country to another.

The Arabic language has its own script with 28 alphabet letters, and some letters have different shapes according to their position in the word. The Arabic text is written from right to left, and there is no capitalization for letters. The Arabic language includes three main parts of speech: noun, verb, and particle [5].

Arabic natural language and text mining applications must deal with several complex problems pertinent to the nature and structure of the Arabic language. For example, the tokenization process for the Arabic text is not a straightforward job because the language is morphologically rich and the words are compact, where a word can correspond to an entire phrase or sentence.

For these reasons, the Arabic language needs careful preprocessing since it has some features that are different from other languages. Besides, these challenges may affect the results of any text analysis process such as classification or sentiment analysis [6].

3. Arabic text categorization

In recent years, and with the rapid increase in the size of information on the Web, text categorization has attracted the attention of many researchers to use the text categorization (classification) as a way to simplify the access to useful information. Text categorization as one of the main text mining tasks can be defined as the process of assigning a predefined category (label) to an unlabeled document (text) based on its content [7]. Text categorization has been used for several applications such as improving the performance of information retrieval systems, as spam filtering, and in medical information systems [8].

In practice, the typical text categorization system includes four main phases, where each phase may further include several other steps. These phases include the text preprocessing phase, which has several steps that aim to prepare the text to be ready to be processed by the categorization model. Usually, this phase includes some other classic natural language processing techniques such as parsing, tokenization, stop word removal, and term weighting, stemming, and part of speech tagging.

Text tokenization is the process of breaking up a sequence of strings into pieces such as words, keywords, phrases, symbols, and other elements called tokens. The result of tokenization process is tokens or terms which make it easy to count the number of the same terms in each document or any other calculation process required for calculating the weight of each term in each document.

Stop word removal is the process of removing punctuation marks, formatting tags, digits, prepositions, pronouns, conjunction, and auxiliary verbs. Stop words can be defined as a set of words that contains high frequency in the document collection and will not help us in the categorization process. The removal of stop words has many advantages, such as reducing the size of the document collection and allowing for deciding the most frequent term in each document. The Arabic language includes several types of stop words. Besides, general words, numbers, and symbols and special characters are also considered as stop words in several applications. Arabic text may also contain words, letters, or sentences from another language such as English. Most applications removed them from the text before processing.

Stemming is defined as the process that returns the segment of the word left after removing some prefixes and suffixes from the word. This process is usually used to reduce the size of the term set of each document in the collection of training documents used in the categorization process [5].

Term selection is defined as the process of selecting the best terms to reduce the term space. In term selection, each term is assigned a weight value based on a weighting scheme in a given text, and then the text is represented as a vector of the weighted terms, and this weight value is used to select the most important terms.

Term weighting: in this process, the terms in the text are weighted using the vector space model. In the vector space model, each term in the text is assigned a weight value based on a method called term frequency-inverse document frequency (TF-IDF). This method is considered one of the most popular methods used to compute term weights [9]. In term frequency-inverse document frequency method, the TF refers to the number of occurrences of term i in a document j, whereas IDF can be computed as follows:

$$IDF_{i,j} = \log\left(\frac{N}{n_i}\right) \tag{1}$$

where N is the total number of documents in the collection; n_i is the number of documents in the collection that contain the term i.

After the term frequency (TF) and inverse document frequency (IDF) are computed for each term, the term weight ($W_{i,j}$) is obtained using the following equation:

$$W_{i,j} = TF_{i,j} \times IDF_{i,j} \tag{2}$$

3.1 Some related works for Arabic text categorization

In the last few years, the importance of text categorization for the Arabic language has attracted many researchers. In this section, we just briefly review some of them. For example, and starting from 2006, some works have used the K-nearest neighbor method for the categorization of Arabic text [10, 11]. Duwairi in [12] compared the three classifiers: Naïve Bayes (NB), K-nearest neighbor (K-NN), and distance-based algorithms for categorizing Arabic text; then Mesleh in [13] proposed an approach using support vector machine (SVM).

Duwairi et al. [14] applied the K-nearest neighbor algorithm for the categorization of Arabic text to compare among three stemming techniques: full stemming, light stemming, and word clusters.

Both Thabtah et al. [15] Noaman et al. [16] proposed approaches based on Naïve Bayesian method for Arabic text categorization. For this purpose, the authors used chi-square for feature selection and Naïve Bayesian for categorization. Gharib et al. [17] applied a support vector machine (SVM) to deal with the problem.

In 2011, several researches were published dealing with Arabic text categorization, and some of them used the same classical methods used before but for different datasets [18] and using different preprocessing methods such as stemming [19].

Some researchers provide a comparative study of some existing approaches. For example, the work presented in [20] compares the sequential minimal optimization (SMO), Naïve Bayesian, and decision tree (C4.5) algorithms to find the most applicable method for the classification of the stemmed and the non-stemmed Arabic text. Both [21] and [22] have proposed Hybrid approaches proposed that combined the K-nearest neighbor method and the binary particle swarm optimization. In the same year, an interesting approach based on association rule mining is proposed by Al-Radaideh et al. in [1].

In 2012, a neural network-based method was proposed by [23]. The authors used learning vector quantization (LVQ) algorithm and Kohonen self-organizing map (SOM) to find similarities in text for the purpose of categorization.

The researches continue proposing different methods to handle the categorization problem of the Arabic text. In 2015, a survey and a comparative study of

several approaches were presented by [24]. Besides, Al-Radaideh and Al-Khateeb [8] applied the associative classifier to classifying Arabic articles from the medical domain. The experimental results reported by the authors showed that the associative classification approach outperformed the Ripper, SVM, C4.5 algorithms.

One of the most frequent challenges in automatic text categorization is the high dimensionality of terms that may affect the performance of the categorization model. A good solution to overcome this challenge is to use feature selection/reduction methods that allow selecting a subset of terms that best represent the whole text in the best way. Ghareb et al. [25] tackled the feature selection part of the text categorization problem where they presented a hybrid feature selection approach that combines several feature selection methods with an enhanced version of the genetic algorithms (GA). In the same direction, the authors of [2] proposed three enhanced filter feature selection methods (Category Relevant Feature Measure, Modified Category Discriminated Measure, and Odd Ratio2) for text classification.

3.2 Case study

One of the recent approaches that tackled the dimensionality reduction issue is proposed by Al-Radaideh and AlAbrat [26]. They proposed a method that used the term weighting and the reduct concept of the rough set theory to reduce the number of terms used to generate the classification rules that represent the rough set classifier.

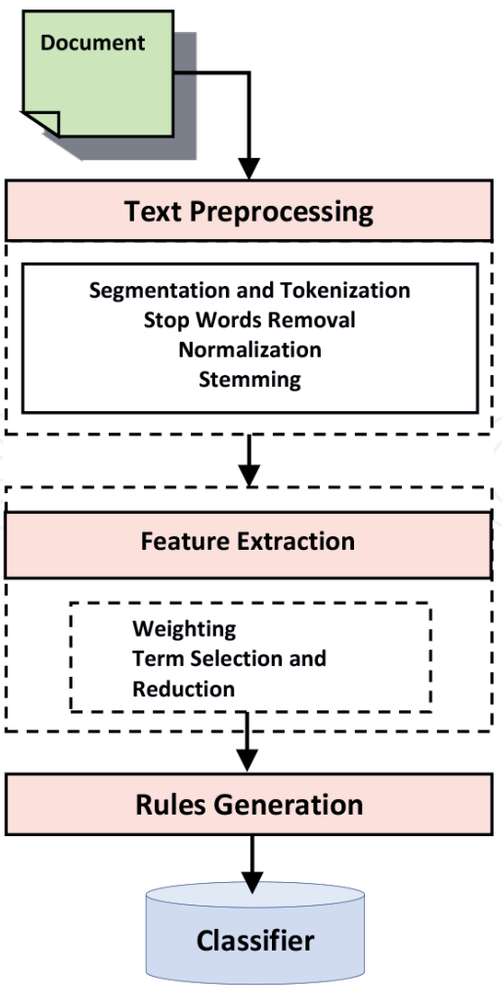

Figure 1.
The main phases of the classic text categorization process.

Category	F-measure for single reduct (%)	F-measure for multiple reducts (%)
Art	89	97
Economy	84	93
Health	94	98
Law	80	89
Literature	80	93
Politics	80	90
Religion	88	92
Sport	94	99
Technology	87	95

Table 1.
The F-measure for the two approaches.

The classification process starts with the classic preprocessing phase, which includes the tokenization step, stop word removal, term weighting, term stemming, and term selection. The main phases of the text categorization process are presented in **Figure 1**. The reduct concept was used to reduce the number of selected terms. The next phase is the rule generation phase. For this purpose, the authors enhanced the quick reduct method and proposed a multiple minimal reduct extraction method. The generated multiple reducts were used to generate the set of classification rules which represent the rough set classifier.

For evaluation purposes, an Arabic corpus of 2700 documents and 9 categories has been used. The documents in the corpus have been categorized manually by human experts into nine categories (art, economy, health, law, literature, politics, religion, sport, and technology). Documents in the corpus were evenly distributed on the 9 mentioned categories, with each having 300 documents.

Two experiments were reported, the first using single minimal reduct and the second using multiple reducts. The F-measure metric for the two methods is presented in **Table 1**. In addition, the reported results showed that the proposed approach had achieved an overall categorization accuracy of 94% when using multiple reducts, which outperformed the single-reduct method which achieved an accuracy of 86%.

4. Arabic sentiment analysis

The growth in the use of the social network's applications such as Facebook, Instagram, and Twitter has emerged into a huge volume of user reviews and opinions about the particular aspects of products or services where people like to share their opinions, feelings, experiences, thoughts, and preferences about these services, products, or events [27]. Reviewers express their feeling according to their understanding and observation.

In practice, this opinion can be used for identifying user interest and trends. One may have a positive opinion, while some others may have a negative opinion at the same time regarding a particular event or a service, and some express a neutral feeling.

Sentiment analysis is considered as one of the recent applications of text categorization that categorize the emotions expressed in a text [27]. To handle the huge amount of textual reviews and opinions, an intelligent automatic method is needed to analyze this huge content and classify it as positive, negative, or neutral opinion. In literature, this automatic process is called sentiment analysis (SA). Sentiment analysis is defined as the task that identifies the polarity of the review or the writer's

attitude toward a particular topic, product, or a service is positive, negative, or neutral [28]. In recent years, this topic attracted many organizations where they use sentiment analysis to enhance the customer relationship management, improve the marketing process, and provide the client's feedback.

In practice, to analyze and classify sentiments of people is a difficult task to handle. There are several reasons for this issue. For example, the shared reviews and feelings are usually not in a structured format and written in a nonstandard language. Therefore, this analysis requires special techniques and semantic algorithms [29].

The science behind sentiment analysis is based on algorithms that use natural language processing concepts to categorize pieces of written text. The algorithms are usually designed to identify positive and negative words, such as "beautiful," "fantastic," "disappointing," and "terrible." There are mainly two levels of sentiment analysis, document level and sentence level.

Most sentiment analyzers work by one or a hybrid of the following four main approaches in scoring a sentence or document for the sentiment [30]:

1. Lexicon-based methods. These methods consider a lexicon dictionary for identifying the polarity of the text. Some predefined lists of positive and negative words were stored with their predefined scores (weights). The methods search for important keywords (usually verbs and adjectives) along with modifiers like negation words.

2. Rule-based methods. These methods look at the presence of some vocabulary words in sentences and use a set of predefined manually crafted rules to categorize these sentences by sentiment.

3. Machine learning methods. These methods treat the problem of sentiment analysis as a classification problem. The methods require some annotated datasets for training. The annotated datasets are lists of texts with sentiments manually recognized. If we take a massive amount of such texts and feed some machine learning algorithms (like Naïve Bayes, neural networks, SVM, deep learning) with them, it will learn how to identify or predict sentiment automatically. The machine learning methods usually involve some steps such as feature extraction from text and training and prediction processes.

4. Combined/hybrid methods. These methods use a lexicon dictionary along with pre-labeled dataset for developing a classification model using some machine learning approaches. Usually, by combining two approaches, the methods can improve the accuracy and precision of the sentiment analysis process.

4.1 Some related works to Arabic sentiment analysis

The majority of work on sentiment analysis has mainly targeted English text, whereas other languages such as the Arabic language have not received enough attention and focus. In recent years, several approaches were developed for Arabic sentiment analysis, where different works were started using different feature selection techniques and different dataset types from several different domains. Here we review some of these efforts.

For example, Abdul-Mageed et al. in [31] proposed a supervised machine learning system for sentiment analysis. They investigated different lexicon information to be either lemma or lexeme. The system used a sentence-level sentiment analysis. In the same year, Shoukry and Rafea [32] investigated using two classical machine learning methods: the Naïve Bayes and support vector machine classifiers.

In 2013, Al-Kabi et al. [33] investigated different classification methods for sentiment analysis. The methods include support vector machine and the Naïve Bayes. The TF-IDF term weighting method is used in the preprocessing step. Abdulla et al. in [34] investigated two main approaches, the corpus approach and the lexicon approach, by manually building the annotated corpus from the art domain.

Al-Subaihin and Al-Khalifa [35] provided a sentiment analysis system for Arabic text. Two algorithms have been designed to analyze and classify the text for their sentiment polarity. The system allows the users to annotate large Arabic text corpus using a game component. Then the system uses the sentiment analyzer component to classify the polarity of the text.

In 2014, some researchers investigated the effects of stemming, feature correlation, and n-gram models for Arabic sentiment analysis [6]. For the classification purpose, three classical classifiers were used, Naive Bayes, K-nearest neighbor, and SVM. In the same direction, Al-Radaideh and Twaiq [36] proposed to use the reduct concept of rough set theory as a term reduction method for the sentiment analysis of Arabic tweets. Both the number of the generated rules and then the number of reducts were used to measure the performance of the proposed method.

In 2015, a supervised ensemble-based classifier proposed by [37] combined different types of features such as opinion and discourse features. The classifier combines the three classifiers (MaxEnt, ANN, and SVM) with the majority voting for rules matching.

The standard Arabic corpus was always a problem for the researchers in the sentiment analysis domain. A prototype Arabic corpus was proposed, in 2016, by Al-Kabi et al. in [38]. The corpus could be used for sentiment analysis, Arabic reviews, and comments. The corpus consists of 250 topics related to 5 predefined domains.

Recently in 2017, Al-Radaideh and Al-Qudah [28] introduced how to use the concepts of rough set theory for term selection for sentiment analysis. The work considered an extension to work presented in [36]. The presented study used compared four reduct generation approaches and two rule generation methods. The results of the experiments showed that using rough set reduct techniques lead to different results, and some of them can perform better than non-rough set classifier. The conclusion of the work indicates that using the concepts of rough set theory for term selection/reduction can achieve good results.

Interesting work is presented by [39] where the authors used a standard corpus and made a review and performance comparison for some of the state-of-the-art approaches used in the multilingual sentiment analysis. One of the surprise results of the study is that the output accuracy of the reviewed approaches using the used standard corpus is far lower than the accuracies reported by the original research. This was justified due to the lack of details in the published works, which did not allow for exact reproduction of the reviewed methods. In some cases, the reported results by the original authors are not comparable with the results reported by [39] because they used different tools, experiment settings, and corpus.

After reviewing the results presented in some of the reviewed work, we could notice that no stable results were reported for the same methods used in the sentiment analysis process. This could be justified that the authors used different Arabic corpora with different dialectics and different stemming techniques and stop word lists.

5. Arabic text summarization

Text summarization is the process of producing a shorter version of a specific text. The main goal of automatic text summarization is condensing the source text into a shorter version (summary) preserving its information content and overall

meaning. In practice, the process aims to extract the most important text segments (called summary) found in a single document or a set of related documents.

To generate a perfect summary and still convey the important information in the original text, a full understanding of the original full documents is critical. To understand the document is considered a hard task since it requires semantic and morphological analysis [3].

5.1 Categories of text summarization approaches

Text summarization methods can be categorized based on different criteria. These categories are summarized as follows:

1. The first category is based on the input of the summarization process. The category includes two types, single- and multi-document summarization. In the single-document summarization, only one document is used as an input to the text summarization process. The work of Suneetha and Fatima [40] is an example of this type. In the multi-document summarization, the input is more than one document (a cluster of related text documents), and the output is a single summary. The work of Kumar and Salim [41] is an example of this type.

2. The second category is based on the output of the summarization process, and here also we have two approaches, extractive and abstractive. In the extractive approach, the extractive summary is created by copying significant units (usually sentences) of the original document. The extractive approach involves two main steps: in the first step, the sentences are assigned some scores on how important they are, while in the second step, the highest-ranking sentences are extracted and presented as a summary. An example of this type is the work of Gupta and Lehal [42].

   In the abstractive approach, an abstractive summary relies on the idea of understanding the original text and retelling it in fewer words; a new vocabulary is added, or even novel sentences are unseen in the original sources. Abstractive approaches require deep natural processing such as semantic representation, compression of sentences, and the reformulation and use of natural language generation techniques, which did not yet reach a mature stage today. An example of this type is the work of Lloret and Palomar [43].

3. The third category is based on the content of the text, and here, two subcategories were found, indicative summary and informative summary. The indicative summary does not substitute the source document; it presents an indication about the document purpose and approach to the user so that the reader can choose which of the original documents to read further. An example of this type is the work presented in [44]. The informative summary can substitute the original document(s) by including informative facts that were reported in the original document. The work of [45] is an example of this type.

4. The fourth category is based on the generality of the summary, where two subcategories were found, generic and query-based summary. In the generic approach, the summarization system creates a summary of a document or a set of documents taking into account all the information found in the document. The work introduced in [46] is an example of this type. For the query-based summary, which is also known as user-focused or topic-focused, user-oriented

summarization systems try to build a summary relevant only to the user query. The work introduced in [47] is an example of this type of summarization.

5. The last category is based on the purpose of the summarization process. The category concerns the possible uses of the summary and the potential readers of the summary. It is divided into two types: domain-independent and domain-dependent. Domain-independent approaches do not have any predefined limitations and can accept different types of text. This kind of summarization makes few assumptions about the audience or the goal for generating the summary; anyone may end up reading the summary. An example of this type is the work in [48]. For the domain-specific approach, the system only summarizes documents belonging to a specific domain such as the medical domain; these kinds of systems exert some limitations on the subject of documents. Such systems know everything about a special domain and use this information for summarization. An example of this type is the work of Al-Radaideh and Bataineh [3].

5.2 The importance and usage of text summarization

Automatic text summarization systems are very significant in various domains such as news, medical, oil and gas, legal, and political domains. A good example, where text summarization could be used in the political domain, is the work presented in [49]. The authors introduced an automatic summarization system that summarizes the proceedings of the European Parliament using word clouds.

In general, text summarization is a very important text analysis and processing task and can be used for several purposes and several tasks. These can be summarized in the following points:

1. Text summarization allows users to quickly find the specific information they are looking for within documents.

2. It is important for different applications in natural language processing (NLP) such as information retrieval, question answering, and text classification systems. These applications can save time and resources, having their actual input text in condensed forms [50].

3. It has become very important for assisting and interpreting text information in today's fast-growing information age. This is presented in many applications for summarization such as summarizing news to short message services (SMS) or wireless application protocol format (WAP) for mobile phone, email summary, government officials' messages, or information for people in business [42].

4. Search engines such as Google uses text summarization to present compressed descriptions of the search results, to help users to decide the relevant documents quickly [43].

5. Summarizing domain-specific text such as news articles and political documents can lead to several benefits, such as:

   - It allows people who are interested in the political domain such as politicians to use text summarization in making decisions and getting the needed information instead of reading the whole document.

- Journalists in newspapers and electronic media take a long time in the preparation of news reports and articles due to a large number of archival and online documents related to the subject.
- Web users who are browsing newspapers and electronic media are overwhelmed with political news every day. Text summarization can save their time by helping them to decide which news to read.
- Mobile companies can use text summarization to summarize urgent news and events to SMS or WAP format and text them to their clients to keep them up to date with the latest news.
- Workers in the news agencies can prepare news briefs with the help of the text summarization.
- Producers of television programs and shows can use text summarization in making their stories and reports.

5.3 Some work in text summarization

The work on text summarization started a long time ago and still attracts many researchers. Several text summarization methods for English and other languages have been proposed in the literature. Over time, attention has shifted from summarizing scientific articles to news articles, electronic mail messages, advertisements, and sentiment-based text.

Research in automatic text summarization has gained importance with the widespread use of the Internet and the rapid increase of online information. Over the years, numerous automatic text summarization approaches have been proposed for English and other languages. For example, Silla Jr. et al. [51] tackled the automatic text summarization task as a classification problem. They used machine learning-oriented classification methods to produce summaries for documents based on a set of attributes describing those documents. Another example is the work presented in [52], where the authors proposed a summarization approach based on GA for sentence extraction. The approach used GA to produce a good summary that is readable, cohesive, and similar to the topic of the document.

Recent research focus has shifted to domain-specific summarization techniques that utilize the available knowledge specific to the domain of text [53] where automatic text summarization techniques have been applied to various domains such as medical, political, news, and legal domains proving that adapting domain-relevant features could improve the summarization performance. For example, Chen et al. [54] have proposed an automated text summarization approach (known as AutoTextSumm) to summarize oil and gas drilling articles. The approach combines statistical features, domain keywords, synonym words, and sentence position to extract the most important points in the document.

Some researchers tackled the multilingual text summarization. For example, Litvak et al. [55] have proposed a single-document extractive summarization called MUSE (multilingual sentence extractor) to improve multilingual summarization. The authors applied genetic algorithms to define a perfect weighted linear combination among 31 text-scoring techniques.

Nandhini and Balasundaram [56] have proposed a genetic-based text summarization system to assist in reading difficulties. The system considers informative score, readability score, and sentence similarity scores to weight the sentence of the text.

5.4 Arabic summarization methods

The first Arabic text summarization system called (Lakhas) has been proposed and implemented by [57]. This system uses some sentence scoring features such as term frequency, sentence position, words in the title, and cue words. A weighted linear combination of these features is used to score sentences. The top-ranked sentences are extracted to form the final summary.

A query-based extractive summarization approach has been proposed by [58]. The approach makes use of the phrasal decomposition of the text where each sentence is ascribed a scoring function that is used to identify the most relevant sentences in the text.

Al-Radaideh and Afif [59] proposed an approach that depends mainly on nouns to indicate the importance of the sentence. The approach was originally proposed for the Korean language.

Sobh in [60] proposed an Arabic extractive text summarization system based on machine learning, which integrates Bayesian classifier and genetic programming (GP) classifier in an optimized way to extract the summary sentences.

Hammo et al. [61] proposed a hybrid approach for Arabic text summarization. The approach used heuristic methods to rank text segments by assigning weighted scores to text segments. They used the Arabic WordNet to identify the thematic structure of the input text to select the most relevant sentences.

Two extractive approaches were proposed by [62]. The first is a graph-based approach, where the text is represented as a graph. The sentences represent the nodes of the graph, and edges between nodes represent the similarity between sentences. The second is a hybrid approach that combines the first approach with a statistical-based model.

Imam et al. [63] proposed an ontology-based summarization system for Arabic document. The system is a query-based system where the generated summaries aim at the user's interest according to the user query.

Oufaida et al. [64] proposed a summarization technique based on a statistical approach for assigning scores, to get minimum redundant and maximum relevance terms. Al-khawaldeh and Samawi [65] proposed an Arabic text summarization approach based on lexical cohesion and text entailment (LCEAS).

Al-Taani and Al-Rousan [66] proposed a multi-document text summarization approach based on clustering techniques. The clustering depends on text semantic to extract relationship across the sentences in a group of related documents.

5.5 Case study

One of the recent proposed approaches is presented by Al-Radaideh and Bataineh [3]. The authors proposed a hybrid, single-document text summarization approach that incorporates domain knowledge, statistical features, and genetic algorithms to extract important sentences from the Arabic text. The experimented domain was the political domain. The approach is tested using two corpora, the KALIMAT corpus and Essex Arabic Summaries Corpus (EASC).

KALIMAT corpus is a multipurpose Arabic corpus which contains 20,291 Arabic articles that were collected from the *Al Watan* newspaper. The articles in this corpus are divided into different topics, including political newswires. For each article there are two different summaries summarized by two human experts. The Essex Arabic Summaries Corpus (EASC) contains 153 Arabic articles collected from Wikipedia and *Al Ra'i* and *Al Watan* newspapers. The dataset contains 10 main topics including politics. For each document, five model extractive summaries are available.

The approach followed the typical main phases and steps of summarization systems. These phases are depicted in **Figure 2**. It can be noticed that some of the steps such as tokenization and stemming are general steps and are used across most text mining tasks such as text categorization and sentiment analysis.

In the approach of [3], the summarization process starts by passing the text into some preprocessing steps, which include text tokenization, stemming, and part of speech tagging. To evaluate and give a score to a sentence (*si*) that may appear in the final summary, the approach used several metrics. These metrics include the domain knowledge score (*Dkw*), term frequency (*TF*), sentence length (*SLen*), sentence position (*SPos*), and sentence similarity to the document's title (*SSTitle*). This score is called the informative score of the sentence which is used to determine its importance.

$$Score\ (si) = Dkw\ (si) + TF\ (si) + SLen\ (si) + SPos\ (si) + SSTitle\ (si). \quad (3)$$

Dkw (*si*) is the score assigned to a sentence based on the number of political keywords it contains by summing up the weights assigned to the keywords in the domain knowledge base.

In the summary generation phase, the final summary is generated using the informative cores, semantic similarity, and genetic algorithms. To extract a cohesive summary from the text, the approach used the cosine similar to discover how sentences are related to each other.

The Recall-Oriented Understudy for Gisting Evaluation (ROUGE) metric was used to compare the automatically generated summaries by the approach against the summaries generated by humans. The ROUGE metric relies on different

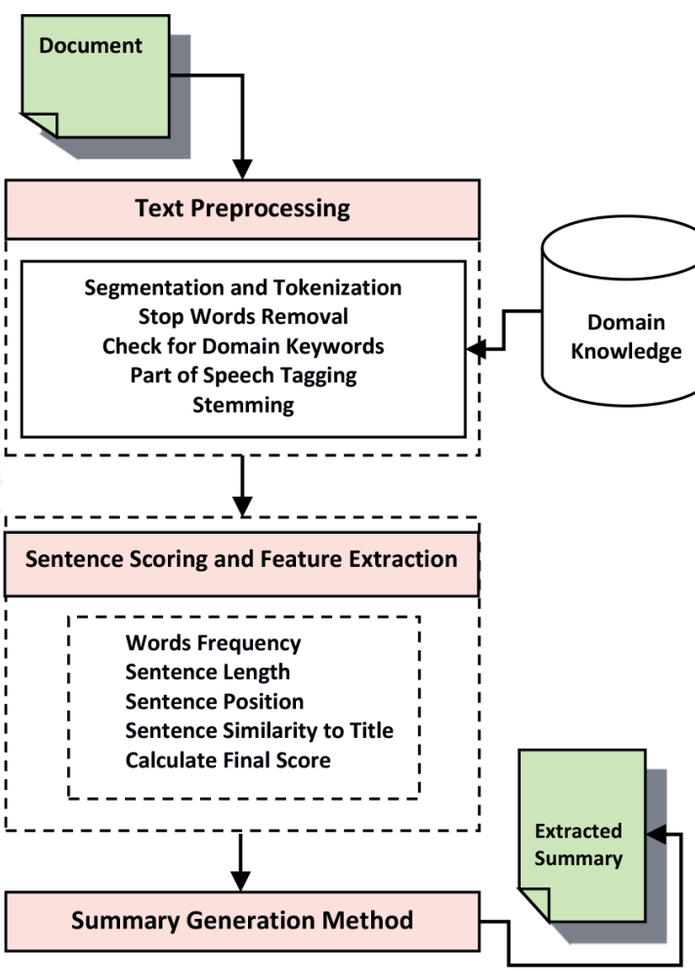

Figure 2.
The phases of the summarization process.

	Without domain knowledge		Using domain knowledge	
	CR = 25%	**CR = 40%**	**CR = 25%**	**CR = 40%**
Recall	36.6%	59.6%	31.9%	50.3%
Precision	62.6%	57.5%	60.7%	56.1%
F-measure	45.8%	60.5%	41.4%	52.8%

Table 2.
The ROUGE-1 evaluation results of the approach.

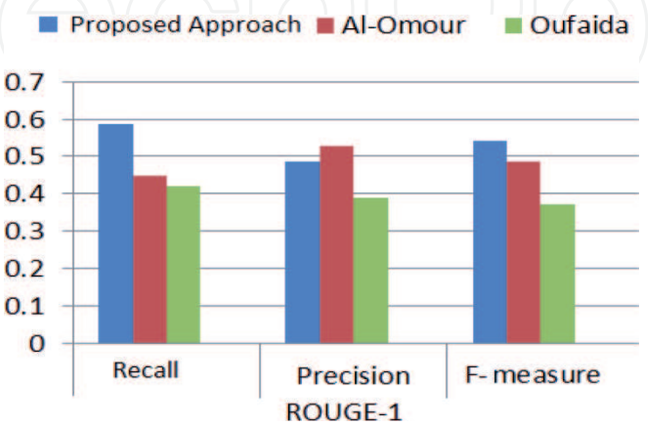

Figure 3.
Results of the three approaches.

types of n-gram co-occurrence. ROUGE-1 and ROUGE-2 compute the number of overlapping unigrams and bigrams, respectively, between the summary extracted by the summarization system and the golden summary created by humans. For each one of these metrics, the recall (R), precision (P), and F-measure (F) were calculated. Recall (R) is a measure of the coverage of the system (completeness). Precision (P) is a measure of correctness to find out how many sentences that the system returns are correct. F-measure (F) is calculated based on precision and recall to provide a single measurement for a summarizer. All the formulas are found in [3].

The approach is tested using two different compression ratios 25% and 40%. The approach is tested using the KALIMAT corpus with and without using the domain knowledge. The results are presented in **Table 2**. It can be noticed from the results presented in **Table 2** that the proposed approach using the domain knowledge achieved higher recall, precision, and F-measure values at both 25% and 40% compression ratios. The approach demonstrated promising results when summarizing Arabic political documents with an average F-measure of 60.5% at the compression ratio of 40%. This indicates the effect of using the domain knowledge corpus in the summarization process.

In the end, the approach is compared against two other Arabic text summarization approaches [62, 64] at a compression ratio of 40%. The results are presented in **Figure 3**.

6. Conclusion

This chapter discussed the three main text mining applications for Arabic language. These applications are text categorization, sentiment analysis, and text summarization. The chapter also reviewed some of the main related works to

these applications and discusses some cases in detail. As a conclusion, although the process of Arabic text categorization was adopted by many researchers who have implemented several categorization algorithms, this field still needs more efforts to be enriched with new and improved algorithms. Till now, most proposed approaches used some well-known methods that could be used for different languages. In the future, new approaches that use the features of Arabic language need to be investigated. A real implementation of a complete Arabic text classification system is still a challenge.

As conclusion for sentiment analysis applications, we can say that some challenges still face the researches in this domain and need to be tackled. These challenges may apply to several languages, including Arabic. Sentiment analysis systems cannot understand perfectly the complexities of the human language. Recognizing context and tone is a difficult process for a machine. Beside, sentiment analysis is still rather incompetent in measuring things such as sarcasm, skepticism, hope, anxiety, or excitement.

Another challenge is that sentiment analysis needs to move beyond a one-dimensional positive to negative scale because there are other kinds of sentiment that cannot be placed on a simple scale. We need a multidimensional scale to truly understand and capture the broad range of emotions that humans express. The last challenge that should be noted is that sentiment analysis is highly domain centered. This means that a developed solution for one domain (e.g., mobile phones) will not directly work on other domains (e.g., hotels). The phrases and patterns used to express sentiment vary across domains and need to be adapted when switching between domains.

As for Arabic text summarization, it is still as one of the open, challenging areas for research in natural language processing (NLP) and text mining fields. Despite the existence of plenty of research work in the domain-based summarization in English and other languages, there is a lack of such work in Arabic due to the shortage of existing knowledge bases. This should motivate the researchers to develop automatic summarization approaches and applications to handle the increasing amount of electronic Arabic documents.

Author details

Qasem Al-Radaideh
Yarmouk University, Irbid, Jordan

*Address all correspondence to: qasemr@yu.edu.jo

References

[1] Al-Radaideh Q, Al-Shawakfa E, Ghareb A, Abu Salem H. An approach for Arabic text categorization using association rule mining. International Journal of Computer Processing of Languages. 2011;**23**(1):81-106

[2] Ghareb A, Bakar AA, Al-Radaideh Q, Hamdan A. Enhanced filter feature selection methods for Arabic text categorization. International Journal of Information Retrieval Research. 2018;**8**(2):1-24

[3] Al-Radaideh Q, Bataineh D. A hybrid approach for Arabic text summarization using domain knowledge and genetic algorithms. Cognitive Computation. 2018;**10**(4):651-669. DOI: 10.1007/s12559-018-9547-z

[4] Farghaly A, Shaalan K. Arabic natural language processing: Challenges and solutions. ACM Transactions on Asian and Low-Resource Language Information Processing. 2009;**8**:22. DOI: 10.1145/1644879.1644881

[5] Al-Kaabi M, Al-Radaideh Q, Akawi K. Benchmarking and assessing the performance of Arabic stemmers. Journal of Information Science (JIS). 2011;**37**(2):111-119

[6] Duwairi R, El-Orfali M. A study of the effects of preprocessing strategies on sentiment analysis for Arabic text. Journal of Information Science. 2014;**40**(4):501-513

[7] Lam W, Ruiz M, Srinivasan P. Automatic text categorization and its application to text retrieval. IEEE Transactions on Knowledge and Data Engineering. 1999;**11**(6):865-879

[8] Al-Radaideh Q, Al-Khateeb S. An associative rule-based classifier for Arabic medical text. International Journal of Knowledge Engineering and Data Mining. 2015;**3**(3-4):255-273

[9] Wang N, Wang P, Zhang B. An improved TF–IDF weights function based on information theory. In: Proceedings of the International Conference on Computer and Communication Technologies in Agriculture Engineering. 2010. pp. 439-441

[10] Al-Shalabi R, Kanaan G, Gharaibeh M. Arabic text categorization using KNN algorithm. In: Proceedings of the 4th International Multi-conference on Computer Science and Information Technology. Jordan: Amman; 2006

[11] Syiam MM, Fayed ZT, Habib MB. An intelligent system for Arabic text categorization. International Journal of Intelligent Computing and Information Sciences. 2006;**6**(1):1-19

[12] Duwairi R. Arabic text categorization. International Arab Journal of Information Technology. 2007;**4**(2):125-131

[13] Mesleh A. Chi-square feature extraction based SVMs Arabic language text categorization system. Journal of Computer Science. 2007;**3**(6):430-435

[14] Duwairi R, Al-Refai M, Khasawneh N. Feature reduction techniques for Arabic text categorization. Journal of the American Society for Information Science. 2009;**60**(11):2347-2352

[15] Thabtah F, Eljinini M, Zamzeer M, Hadi W. Naïve Bayesian based on chi-square to categorize Arabic data. In: Proceedings of the 11th International Business Information Management Association Conference (IBIMA) Conference on Innovation and Knowledge Management in Twin Track Economies, Cairo. 2009. pp. 930-935

[16] Noaman H, Elmougy S, Ghoneim A, Hamza T. Naïve Bayes classifier based Arabic document categorization. In: In: Proceedings of the 7th International Conference in Informatics and Systems (INFOS 2010); Cairo, Egypt. 2010

[17] Gharib TF, Habib MB, Fayed ZT. Arabic text classification using support vector machines. International Journal of Computers and Applications. 2009;**16**(4):1-8

[18] Al-Salemi B, Aziz M. Statistical Bayesian learning for automatic Arabic text categorization. Journal of Computer Science. 2011;**7**(1):39-45

[19] Wahbeh A, Al-Kabi M, Al-Radaideh Q, Al-Shawakfa E, Alsmadi I. The effect of stemming on Arabic text classification: An empirical study. International Journal of Information Retrieval Research. 2011;**1**(3):54-70

[20] Hussien MI, Olayah F, Al-dwan M, Shamsan A. Arabic text classification using SMO, Naive Bayesian, J48 algorithm. International Journal of Research and Reviews in Applied Sciences. 2011;**9**(2):306-316

[21] Chantar HK, Corne DW. Feature subset selection for Arabic document categorization using BPSO-KNN. In: Nature and Biologically Inspired Computing (NaBIC). 2011. pp. 545-551

[22] Chen Y, Zeng Z, Lu J. Neighborhood rough set reduction with fish swarm algorithm. Soft Computing. 2017;**21**(23):6907-6918

[23] Azara M, Fatayer T, El-Halees A. Arabic text classification using learning vector quantization. In: Proceedings of the 8th International Conference on Informatics and Systems (INFOS2012). 2012. pp. 39-43

[24] Hmeidi I, Al-Ayyoub M, Abdulla N, Almodawar A, Abooraig R, Mahyoub N. Automatic Arabic text categorization: A comprehensive comparative study. Journal of Information Science. 2015;**41**(1):114-124

[25] Ghareb A, Hamdan A, Bakar A. Hybrid feature selection based on enhanced genetic algorithm for text categorization. Expert Systems with Applications. 2016;**49**:31-47

[26] Al-Radaideh Q, AlAbrat M. An Arabic text categorization approach using term weighting and multiple reducts. Journal of Soft Computing. 2018;**2018**:1-15

[27] Rahmath H, Ahmad T. Sentiment analysis techniques–A comparative study. IJCEM International Journal of Computational Engineering & Management. 2014;**4**(17):25-29

[28] Al-Radaideh Q, Al-Qudah G. Application of rough set-based feature selection for Arabic sentiment analysis. Cognitive Computation. 2017;**9**(4):436-445

[29] Kumari U, Soni D, Sharma A. A cognitive study of sentiment analysis techniques and tools: A survey. International Journal of Computer Science and Technology. 2017;**8**(1):58-62

[30] Vohra M, Teraiya J. A comparative study of sentiment analysis techniques. Journal of Information, Knowledge and Research in Computer Engineering. 2013;**2**:313-317

[31] Abdul-Mageed M, Kübler S, Diab M. SAMAR: A system for subjectivity and sentiment analysis of Arabic social media. In: Proceedings of the 3rd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis. 2012. pp. 19-28

[32] Shoukry A, Rafea A. Sentence-level Arabic sentiment analysis. In: Proceedings of International Conference on Collaboration Technologies and Systems (CTS); Denver. 2012. pp. 546-550

[33] Al-Kabi M, Abdulla N, Al-Ayyoub M. An analytical study of Arabic sentiments: Maktoob case study. In: Proceedings of 8th IEEE International Conference on Internet Technology and Secured Transactions (ICITST). 2013. pp. 89-94

[34] Abdulla NA, Ahmed NA, Shehab MA, Al-Ayyoub M. Arabic sentiment analysis: Lexicon-based and corpus-based. In: Proceedings of IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT). 2013. pp. 1-6

[35] Al-Subaihin A, Al-Khalifa H. A system for sentiment analysis of colloquial Arabic using human computation. The Scientific World Journal. 2014;**2014**:8. Article ID: 631394. DOI: 10.1155/2014/631394

[36] Al-Radaideh Q, Twaiq L. Rough set theory approaches for Arabic sentiment classification. In: Proceedings of International Conference on Future of Things and Cloud, IEEE Computer Society. 2014

[37] Bayoudhi A, Hadrich L, Ghorbel B. Sentiment classification of Arabic documents: Experiments with multi-type features and ensemble algorithms. In: Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation; Shanghai, China. 2015. pp. 196-205

[38] Al-Kabi M, Al-Ayyoub M, Alsmadi I, Wahsheh H. A prototype for a standard Arabic sentiment analysis corpus. The International Arab Journal of Information Technology. 2016;**13**(1A):163-170

[39] Dashtipour K, Poria S, Hussain A, Cambria E, Hawalah A, Gelbukh A, et al. Multilingual sentiment analysis: State of the art and independent comparison of techniques. Cognitive Computation. 2016;**8**:757-771

[40] Suneetha M, Fatima S. Corpus based automatic text summarization system with HMM tagger. International Journal of Soft Computing and Engineering (IJSCE). 2011;**1**(3):2231-2307

[41] Kumar Y, Salim N. Automatic multi document summarization approaches. Journal of Computer Science. 2011;**8**(1):133-140

[42] Gupta V, Lehal G. A survey of text summarization extractive techniques. Journal of Emerging Technologies in Web Intelligence. 2010;**2**(3):258-268

[43] Lloret E, Palomar M. Text summarization in progress: A literature review. Artificial Intelligence Review. 2010;**37**(1):1-41

[44] Saggion H, Lapalme G. Generating indicative-informative summaries with SumUM. Computational Linguistics. 2002;**28**(4):497-526

[45] Yih W, Goodman J, Vanderwende L, Suzuki H. Multi-document summarization by maximizing informative content-words. In: Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI); Hyderabad, India. 2007. pp. 1776-1782

[46] Gong Y, Liu X. Generic text summarization using relevance measure and latent semantic analysis. In: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, New Orleans, LA, USA. 2001. pp. 19-25

[47] El-Haj M, Kruschwitz U, Fox C. Experimenting with automatic text summarization for Arabic. In: Vetulani Z, editor. Human Language Technology. Challenges for Computer Science and Linguistics. LTC 2009. Lecture Notes in Computer Science. Vol. 6562. Berlin, Heidelberg: Springer; 2011

[48] Nomoto T, Matsumoto Y. The diversity-based approach to open-domain text summarization. Information Processing & Management. 2003;**39**(3):363-389

[49] De-Hollander G, Marx M. Summarization of meetings using word clouds. In: The Computer Science and Software Engineering (CSSE) CSI International Symposium; Tehran. 2011. pp. 54-61

[50] Pal A, Maiti P, Saha D. An approach to automatic text summarization using simplified Lesk algorithm and Wordnet. International Journal of Control Theory & Computer Modeling (IJCTCM). 2013;**3**(4):15-23

[51] Silla CN, Pappa GL, Freitas AA, Kaestner CAA. Automatic text summarization with genetic algorithm-based attribute selection. In: Lemaître C, Reyes CA, González JA, editors. Advances in Artificial Intelligence—IBERAMIA. Lecture Notes in Computer Science, Vol. 3315. Berlin, Heidelberg: Springer; 2004

[52] Qazvinian V, Hassanabadi L, Halavati R. Summarising text with a genetic algorithm-based sentence extraction. International Journal of Knowledge Management Studies. 2008;**2**(4):426-444

[53] Yeh J, Ke H, Yang W, Meng I. Text summarization using a trainable summarizer and latent semantic analysis. Information Processing and Management. 2005;**41**(1):75-95

[54] Chen Y, Foong O, Yong S, Kurniawan I. Text summarization for oil and gas drilling topic. International Journal of Computer, Electrical, Automation, Control and Information Engineering. 2008;**2**(6):1799-1802

[55] Litvak M, Last M, Friedman M. A new approach to improving multilingual summarization using genetic algorithms. In: The 48th Annual Meeting of the Association for Computational Linguistics; Uppsala, Sweden. 2010. pp. 927-936

[56] Nandhini K, Balasundaram S. Use of genetic algorithms for cohesive summary extraction to assist reading difficulties. Applied Computational Intelligence and Soft Computing. 2013;**2013**:11. Article ID: 945623. DOI: 10.1155/2013/945623

[57] Douzidia F, Lapalme G. Lakhas, an Arabic summarization system. In: The Document Understanding Conference (DUC); Boston, USA. 2004. pp. 128-135

[58] Bawakid A, Oussalah M. A semantic summarization system: The University of Birmingham at TAC 2008. In: The First Text Analysis Conference (TAC); Maryland, USA. 2008. pp. 1-6

[59] Al-Radaideh Q, Afif M. Arabic text summarization using aggregate similarity. In: The International Arab Conference on Information Technology (ACIT'2009); Yemen. 2009. pp. 1-8

[60] Sobh I. An optimized dual classification system for Arabic extractive generic text summarization [M.Sc. thesis]. Giza, Egypt: Department of Computer Engineering, Cairo University; 2009

[61] Hammo B, Abu-Salem H, Evens M. A hybrid Arabic text summarization technique based on text structure and topic identification. International Journal of Computer Processing of Languages. 2011;**23**(01):39-65

[62] Al-Omour M. Extractive-based Arabic text summarization approach [M.Sc. thesis]. Irbid, Jordan: Department of Computer Science, Yarmouk University; 2012

[63] Imam I, Hamouda A, Khalek H. An ontology-based summarization system for Arabic documents (OSSAD).

International Journal of Computers and Applications. 2013;**74**(17):38-43

[64] Oufaida H, Nouali O, Blache P. Minimum redundancy and maximum relevance for single and multi-document Arabic text summarization. Journal of King Saud University Computer and Information Sciences. 2014;**26**(4):450-461

[65] Al-Khawaldeh F, Samawi V. Lexical cohesion and entailment based segmentation for Arabic text summarization (LCEAS). World of Computer Science and Information Technology Journal (WSCIT). 2015;**5**(03):51-60

[66] Al-Taani A, Al-Rousan S. Arabic multi-document text summarization. In: The 17th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing 2016); Turkey. 2016