Journal of Information Technology and Computing
https://jitc.sabapub.com
ISSN: 2709-5916

2025 Volume 5, Issue 1: 1–14
DOI: https://doi.org/10.48185/jitc.v5i1.1702

Paragraphic Query-Based Summarization Approach for Arabic Extracts

Mohammed Salem BinWahlan*, Mazin Alkathiri & Khaled Mohammed Binabdl
Information Technology Dept., College of Computers, Seiyun University, Yemen

Received: 10.07.2024 • Accepted: 24.08.2025 • Published: 14.09.2025 • Final Version: 12.10.2025

Abstract: The internet's exponential increase in information availability imposes a significant cost on individuals seeking such content. This challenge is being addressed by numerous researchers striving to overcome its complexity. The current study presents an alternative approach to query-based automatic text summarization. Unlike earlier methods, this approach generates the query based on the first paragraph sentences of the material being summarized, eliminating the need for user input during query submission. Applying the generated query to the document results in a final summary. Additionally, the study investigates the impact of different query lengths and similarity measures. The evaluation utilized the ROUGE metric and the EASC dataset. The experimental findings show that the suggested approach, which makes use of the Russel similarity measure and a longer query length, performs better than alternative scenarios that use the cosine, Forbes, and Jaccard similarity measures and shorter query lengths.

Keywords: Query, similarity measure, summarization, information retrieval

1. Introduction

The internet's exponential increase in information availability imposes a significant cost on individuals seeking such content. Numerous scholars are working to address this difficulty and get past its complexity. Researchers are concentrating on employing text summarization for these enormous amounts of data to provide answers to this issue. Since the 1950s and 1960s, researchers have actively pursued the area of automatic text summarization [1, 2, 3]. Text summarization aims to condense a lengthy text into a concise form (summary) that includes key information. The resulting summary can be categorized into two types: extractive and abstractive [69]. An extractive summary includes the most crucial text segments from the original document without altering their structure, whereas an abstractive summary involves rephrasing these segments' structures before incorporating them into the summary content. This complexity makes producing abstractive summaries more challenging than extractive summaries.

Automatic text summarization, as an open research problem, has spurred many researchers to investigate and present different solutions in the form of automatic text summarizers. The starting point for all efforts in this area is Luhn's work [1], in which he presented a system based on word and sentence significance. The former is calculated based on the word's appearance in the document, while the latter is calculated using the location in which the sentence's words appear. Later, a variety of text summary techniques—from easy to difficult—

* Corresponding Author: *moham2007med@yahoo.com

were created. According to some studies [4, 5, 6, 8, 9, 10,16, 11, 25] some methods were based on term weight schemes such as Term Frequency-Inverse Document Frequency, which assigns greater weight to a document's most significant content. Afsharizadeh et al. [12] proposed a query based approach that summarizes the text by scoring its units relayed on a dual-feature category: features extracted from a sentence and query. These feature values are adjusted by weights, and the sum of the adjusted feature values is used as the document sentence score. The final summary is composed of the text units that have received the greatest rates. Rani & Lobiyal [13] introduced a text summarization model to handle large datasets and recognize semantic, statistical, and linguistic features. The researchers utilized the TF-IDF weighting scheme, assigning each word in the text a TF-IDF weight and a weighted word vector. They asserted that a more varied summary is produced by their model. ROUGE metric and the DUC 2007 dataset were used for the model's assessment [55].

Using many features, other approaches identify the important document content as a final extract. According to Douzidia & Lapalme [14], Zechner [15], El-Haj et al. [16], Thakkar et al. [22], Binwahlan [19], Teufel [21], Amato et al. [23, 24], Elhaj [18], Andhale & Bewoor [26], Alami et al. [29], and Bhola et al. [27], the text segments are rated as a summation of the scores of those attributes. Verma et al. [28] utilized various techniques to develop a hybrid model for the summarizing problem. The document phrases were grouped according to how similar they were, and the text feature's modified rates of each cluster's key phrases were used to identify which phrases were significant. To determine the phrases' ultimate rates, these rates were fed into a system of fuzzy inference. Three different datasets were used: DUC 2001, DUC 2002, and CNN. The researchers observed that effective summaries were produced by combining clustering, fuzzy, and evolutionary techniques. Singh et al. [30] presented a supervised method for text summarization. Nine text features were used: aggregation score, the count of ‘‘incorrect words’’, thematic words, entropy, numeric data, sentence position, sentence length, named entity, and POS. These are inputted into particle swarm optimization to produce optimal weights. The extracted features are, then, updated by those optimal weights. The final summary contained the sentences from the document that scored the highest after the document sentences were ranked according to adjusted text feature scores.

Machine learning methods aim to extract key features from the content of documents within a given corpus. These extracted features are then utilized to identify the representative content of a document, which is presented as a summary [20, 30, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52]. The performance issues with optimization-based extractive multi-document summarizing techniques are addressed by Wahab et al. [53] with their decomposition-based Multi-Objective Differential Evolution method. The goal of the researchers' work was to tackle this challenge. The evaluation was conducted using the DUC2002 dataset and the ROUGE metric [55]. An artificial text summarizing technique based on an optimization solution was presented by Pati & Rautray [33]. Particle Swarm Optimization (PSO), Butterfly Optimization (BO), and Ant Colony Optimization (ACO) were the three optimization techniques used. Sentence scores were calculated based on term frequency (TF), then the document sentences were scored using cosine similarity measure with TF weighting scheme, resulting in a sentence similarity matrix. This matrix was used as input for the optimization method to suggest summary sentences. The summary sentences were ranked using the Technique for Order Performance by Similarity to Ideal Solution (TOPSIS). The performance of these optimization methods was assessed using the DUC 2006 dataset. The researchers concluded that the Particle Swarm Optimization (PSO) based method outperformed the other methods.

Text summarization and information retrieval share the same goal [31], which is providing users with the information they need. In information retrieval, search engines require a user query for retrieving a group of relevant documents, while in text summarization, summarizers help users know the key content of the retrieved documents [32]. Therefore, information retrieval methods support text summarization in creating more high-quality and concise summaries. Similarity measures and query are two information retrieval ideas that are applied to the text summarization problem in this work. The sentences in the provided text and a query are compared using four similarity metrics, and the target summary contains those most pertinent sentences. The goal of the study is to present a novel method for query-based automatic text summarization. This study avoids user intervention by generating the inquiry from the first paragraph phrases of the material being summarized, in contrast to earlier techniques where the user enters the query. Finding the best similarity metric and query technique to get a more accurate summary is another goal. For assessment, the ROUGE measure [55] is utilized in conjunction with the EASC dataset [54].

The remainder of this document is structured as follows: Part 2 presents relevant work, Part 3 discusses the suggested method, Part 4 presents the results, and Part 5 concludes with recommendations for further research.

2. Related work

This section reviews several works that dealt with similarity measures and queries in text summarizing.

A query-based method of summarizing multi-documents is based on the user's query. Each sentence is scored based on the overlap between the document sentences and the query. The final summary contains those greatest scoring sentences [56, 60]. El-Haj et al. [16] introduced an Arabic text summarization system based on a user-submitted query that uses Salton et al.'s model [57]. This system employs term TF and ISF weight schemes to calculate the matching degree between each sentence and the query. A knowledge builder for extracting multi-word concepts and a summarizer are the two components of the Arabic text query-based summarizing method that Imam et al. [17] suggested. First, the given query is enlarged, and then the document is summarized. Sentences are graded based on how relevant they are to the enlarged and original query. Information extraction and query-based summarizing are combined in Peng et al.'s proposed query-based summarization method for social network messages [58]. Significant information is extracted from social network communications in the extraction process, which results in a feature matrix that is converted into the time-frequency domain. The degree to which each feature's information quality contributes is determined by the expectation-maximization method, or EM method. Sentence weight and query relevance are used to produce the score, which is then used to select the most important sentences for the final summary. A deep auto-encoder (AE) is used to construct term frequency features in a query-based document summarizing method that Yousefi-Azar & Hamey [59] proposed. Based on the query terms entered, Auto-Encoders produce a concept vector for every sentence. The researchers proposed an Ensemble Noisy Auto-Encoder (ENAE) to address the effect of random noise on local TF, which serves as the AE's input representation. A query-based microblogs summarizing technique centered on query relevancy and post time was presented by Geng et al. [62]. To highlight key phrases and temporal information, microblog sentences are clustered. A graph is used to iteratively pick summary sentences, with microblog sentences being ranked using the query-lexical-rank (Q-LexRank) algorithm. Alhoshan & Altwaijry [63] concentrated on query relevance and time information in another investigation. The researchers unveiled a mechanism for updating summaries. This approach works on the assumption that the user already has some information and wants to keep track of any updates pertaining to it. In the summarizing process, semantic and lexical data identified based on the Arabic WordNet lexicon are combined to determine word similarities on a graph.

A well-known thought is that a query-based summarization method cannot generate a summary covering all document content; hence, some key points in the document may be overlooked [60]. The present research's main contribution is to challenge this idea by presenting a technique for creating a query based on the content of the document. Using this method, the query word list is created by adding the terms from each first paragraph sentence. The query word list is then ordered in reverse according to the words' frequency in the document. The final query is then chosen from the words with the highest frequencies.

Similarity measures are considered a main factor in the text summarization process [64, 65]. According to Salim [66], Tanimoto, Forbes, Russell, and cosine are examples of similarity metrics that work well for information retrieval (IR) problems. In Alguliyev et al.'s study [65], three similarity measures (cosine, Jaccard, and overlap) were combined, where cosine and Jaccard similarity measures are symmetric, and overlap is asymmetric. The combined metrics yielded good results. This prompted the researcher of the current study to investigate the behavior of four similarity measures (cosine, Russell, Forbes, and Jaccard/Tanimoto). Sanchez-Gomez et al. [7] examined three term weights and five similarity measures (NGD cosine, RRN, overlap, and Jaccard) for their effectiveness in the text summarization process. Various combinations of similarity metrics and term-weight strategies were showcased. The researchers came to the conclusion that cosine metric and the term-frequency inverse-sentence-frequency scheme worked better than alternative similarity metrics and weight schemes, respectively.

3. The proposed method

The ultimate goal of any summarization method is to generate a suitable representative summary that typifies the most significant content of a document. To achieve this, a query-based Arabic text summarization method is presented. The proposed method goes through four steps to reach its goal: preprocessing, query formulation, similarity calculation, and finally, summary generation. The entire process is illustrated in Figure 1, and each step will be discussed in the following subsections.

3.1. Preprocessing

The resulting summary's quality is directly impacted by how the document's content is handled. As a result, before using the summarizing approach, the original material must undergo some preprocessing, such as tokenization, stop-word removal, and stemming.

3.1.1 Tokenization: this process produces separated text units resulting in the splitting of the source document, in this study, the input document is split into paragraphs (based on a newline character), sentences (using full stops, question marks, and exclamation marks), and words (white space, semicolons, commas, and quotations).

3.1.2 Normalization: a process to combine the various letter shapes into a single, distinctive one and eliminate any odd letters (numbers, special symbols, punctuation marks, and others).

3.1.3 Stop-word removal: a process of eliminating words that have no significant impact on the content is based on consulting Khoja and Garside's stop-word list [67].

3.1.4 Stemming: a process that aims to replace all inflectional or derivational forms with their unique stem. This action leads to the minimization of document terms, resulting in a reduction in processing costs. The stemming step was performed using Khoja and Garside's stemmer [67].

3.2. Query formulation

According to existing research, a user submitting a query based on the final summary produced is necessary for a query-based text summarization [62, 63, 59, 16, 58, 17, 61]. The inquiry in the current study is automatically constructed from the first paragraph sentences in a document, which sets it apart from earlier research. All sentences in the first paragraph are designated for query generation during preprocessing. These sentences' words are inserted into a vector, each word's frequency (TF) is computed, and the unique words along with their TF are added to a query vector. Based on TF weights, the query vector is reorganized in inverse order. The ultimate inquiry is then chosen from the highest weighted, n-sorted words. Algorithm 1 is an illustration of this procedure.

Algorithm 1: Query formulation

INPUT: QL: query length, FTlist = $\{\{t_1, tf_1\}, t_2, tf_2\}, \ldots, \{t_m, tf_m\}\}$//words and the frequency with which they appear in each of a document's first paragraph sentences, where m is the total word count contained in any first paragraph sentence.

OUTPUT: $Q = \{t_1, t_2, \ldots, t_{QL}\}$// query list containing top ranked TF weight terms in D.

1. rank FTlist in inverse order
2. for k ← 1 to QL
3. Q[k] ← FTlist[k]
4. end for

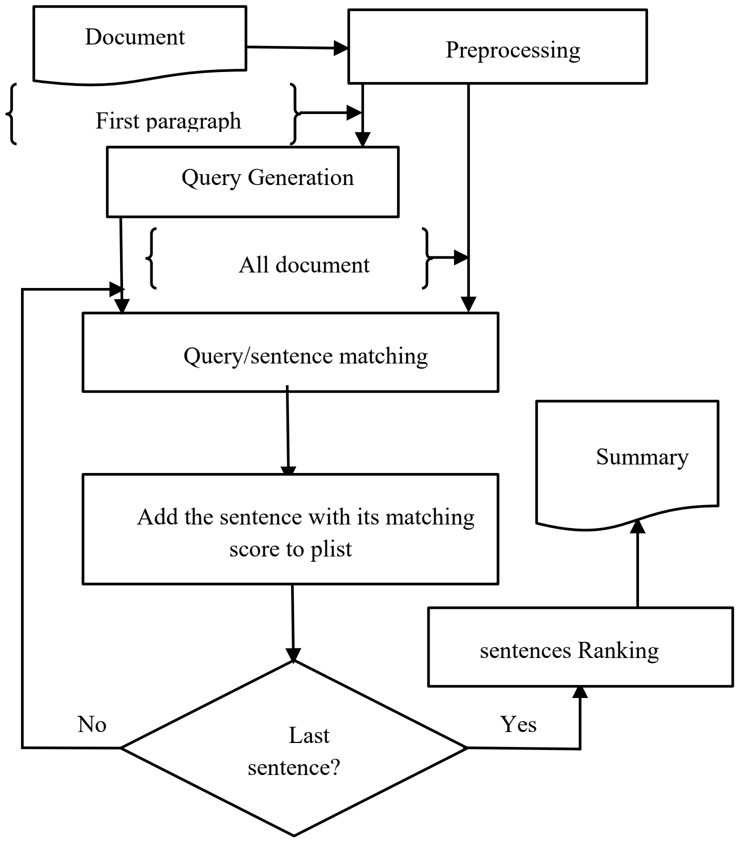

Figure 1. the proposed method architecture

3.3. Similarity measures

A similarity metric can be derived using a formula that assigns a numerical value to a vector pair. In its most basic version, one may look at the word counts that are shared by the document and query vectors. In the current trials, four similarity metrics—Tanimoto, Russell, cosine, and Forbes with tf-isf weighting scheme—are evaluated to determine which is best for the summarization challenge. When calculating the similarity measure, the following processes and structures must be prepared: All unique terms extracted from D (document set) are contained in vector T = $\{t_1, t_2, ..., t_C\}$, C denotes the total terms counts. The weights of terms contained in each document sentence S_q form a vector. S_q is equal to $\{w_{q1}, w_{q2}, ..., w_{qC}\}$. According to Salton and Buckley [68], the word weights are computed using the following equation 1 (tf–isf):

$$w_{qj} \text{ equals } \log(n/n_j) * tf_{qj}. \qquad (1)$$

where n_j is the sentences' number where the j^{th} word appear, n refers to the total document sentences' number, and tf_{qj} is the summation of frequencies of the j^{th} word in S_q.

The sentences in the document are arranged in a vector, where $d_k = \{s_1, s_2, ..., s_n\}$. Like every document sentence, a query term's weights are represented in a vector as well. These vectors are represented in the vector space model as a matrix of size n × C [57]. The similarity metrics [66], displayed in Table 1, are applied based on the produced VSM to ascertain the document sentence's relevance to the inquiry.

Table 1. Similarity measures used for calculating the relevance of a query and a document sentence.

Measure	Formula	
Ochiai/Cosine	$\frac{\sum_{j=1}^{M}(w_{jk}w_{jl})}{\sqrt{\sum_{j=1}^{M}(w_{jk})^2 \sum_{j=1}^{M}(w_{jl})^2}}$	(2)
Russell/Rao	$\frac{\sum_{j=1}^{M} w_{jk}w_{jl}}{n}$	(3)
Forbes	$\frac{n\sum_{j=1}^{M}(w_{jk}w_{jl})}{\sum_{j=1}^{M} w_{jk}^2 \sum_{j=1}^{M} w_{jl}^2}$	(4)
Jaccard/Tanimoto	$\frac{\sum_{j=1}^{M}(w_{jk}w_{jl})}{\sum_{j=1}^{M}(w_{jk})^2+\sum_{j=1}^{M}(w_{jl})^2-\sum_{j=1}^{M}(w_{jk}w_{jl})}$	(5)

3.4. Paragraphic query based Arabic text summarization

As mentioned in subsection 3.3, a vector space model (VSM) is constructed with each row corresponding to a sentence number holding the TF-ISF weights of the terms in that sentence, the leftmost column carrying sentence numbers, and the first row including all terms in the document set. The query word weights are in the final row in VSM. Each document sentence and the query are compared to see how similar they are, using the similarity measures listed in Table 1 one at a time. Every determined resemblance is regarded as a sentence's score. A list containing the scores and sentence numbers is created. The order of this list is reversed. As seen in Figure 1, the final summary is produced by selecting the sentences with the greatest score from the list. Several query length (QL) values are evaluated, including LLS (the length of the longest sentence in the document), 2LLS, 3LLS, and QL equaling the length of the query vector. A similarity degree between each sentence in the document and the query is computed using each query length.

4. Results

The experimental design, datasets utilized, and experimental outcomes are presented in this part, which is based on the experiments conducted in this paper.

4.1. Experimental Design

This subsection discusses the suggested method's experimental design. The dataset is the EASC [54], which consists of 153 documents spanning a number of fascinating topics. Additionally, each source article has five handwritten summaries. The sentences in each article's handwritten summaries are grouped into three categories due to the disagreement among the authors of these summaries regarding which sentences should be included in each summary. The first group, dubbed level 3, includes the sentences that appeared in every human summary; the second group, dubbed level 2, contains the sentences that appeared in any two human summaries; and the third group, dubbed level all, contains the sentences that appeared in any human summaries [18]. Several iterations of the ROUGE evaluation metric [55] are tested for evaluation purposes: while N = 2, ROUGE-N performs well while summarizing a single document. To this purpose, it is employed to evaluate both human and proposed technique summaries, with a 50% length setting [18]. "ROUGE-cut," another ROUGE parameter, is set to 100, which indicates that 100 words are chosen at random from a summary for evaluation. This study implements and evaluates eight experiments: four for each of the four query length values and four for each of the four similarity metrics.

4.2. Experimental results

Based on a produced query, the suggested approach generates a summary. We explore four possible choices of query length (QL): QL = 2LLS, 3LLS, QL = length of the query vector, and QL equals LLS (the longest sentence in the document). These query lengths are used to construct four summaries. Every summary has a strong correlation with the reference (human) summary. Analysis is done on summaries made with a 3LLS query length as a middle length in order to find out how efficient similarity measures are. The ROUGE-2 recall, ROUGE-2 precision, and ROUGE-2 average F findings are shown in Table 2 together with the cosine similarity measure for each of the four query lengths. These findings clearly show that the length of the query has a major influence on the caliber of the summary that is produced. The query that yields the greatest results is QL, which contains terms from every sentence in a document's first paragraph sentence. Conversely, the option with the smallest query length, LLS, produces the least number of results. The outcomes of queries with durations of 2LLS and 3LLS differ slightly from one another. Because QL takes into account all first paragraph sentences, it is preferable to other query lengths. On the other hand, because of its restricted area, LLS loses some first paragraph sentences, which lessens its impact. The comparable effects of query lengths 2LLS and 3LLS suggest that they contain comparable amounts of space, enabling the inclusion of an equal number of first paragraph sentences. Using summaries derived from a query of length 3LLS, Table 3 presents the findings of ROUGE-2 recall, ROUGE-2 precision, and ROUGE-2 average F for the four similarity measures. Russell earns the best results among the similarity metrics, based on the data in this table. Examining the inner workings of these similarity measures, we find that all of them—Russell excluded—divide the computation made in the numerator by a computation of the current sentence's term weights. The computed similarity degree is negatively impacted by each sentence's different denominator value. Russell, on the other hand, divides the calculation in the numerator, fairly, by the total number of document sentences, or n, which has a beneficial effect on the similarity degree that is obtained.

Table 2. Results of ROUGE-2 recall, ROUGE-2 precision and ROUGE-2 average F, for the four different query lengths with cosine similarity measure.

Level	Measure	LLS	2LLS	3LLS	QL
ALL	R	0.68226	0.70103	0.69908	0.71739
	P	0.69115	0.7087	0.70551	0.72318
	Avg	0.68581	0.7041	0.70163	0.7197
2	R	0.63199	0.64629	0.64275	0.6631
	P	0.63316	0.6439	0.64148	0.66077
	Avg	0.63132	0.64354	0.64099	0.66087
3	R	0.54843	0.55271	0.55743	0.57275
	P	0.46038	0.46799	0.46549	0.48119
	Avg	0.48715	0.4948	0.49385	0.51081

Table 3. Results of ROUGE-2 recall, ROUGE-2 precision and ROUGE-2 average F, for the four similarity measures using summaries generated based on a query of length 3LLS.

Level	Measure	Cosine	Forbes	Russell	Jaccard
ALL	R	0.66402	0.51613	0.72102	0.71847
	P	0.67243	0.53306	0.72717	0.72455
	Avg	0.66736	0.52315	0.72344	0.72085
2	R	0.61761	0.50535	0.65715	0.65704
	P	0.61946	0.51487	0.65485	0.65467
	Avg	0.61758	0.50867	0.65506	0.65491
3	R	0.55479	0.47252	0.582	0.5818
	P	0.46664	0.38817	0.48157	0.48198
	Avg	0.4931	0.41208	0.51209	0.5123

5. Conclusion and upcoming work

This work proposes a query-based text summary approach for Arabic texts, using a different query formulation than earlier query-based text summarization approaches. The novel method of query construction does not require user intervention; instead, it builds the query using the first paragraph sentences. The effects of four distinct query length values and four similarity metrics are examined throughout the tests. It has been found that extending the query length produces beneficial results. The Russell similarity metric performs better than the other measures in terms of similarity. Future ideas include thinking about multi-document summarizing as a future attempt and researching more information retrieval strategies for the Arabic text summarization challenge.

References

[1] H. P. Luhn, "The Automatic Creation of Literature Abstracts". *IBM Journal of Research and Development*, 2(92) (1958) 159-165.

[2] H. P. Edmundson, "New Methods in Automatic Extracting". *Journal of the Association for Computing Machinery,* 16(2) (1969) 264-285.

[3] P. Baxendale, "Machine-made Index for Technical Literature - an Experiment". *IBM Journal of Research Development*. 2(4) (1958) 354-361.

[4] Y. T. Sung, K. E. Chang, T. C. Liu, “The effects of integrating mobile devices with teaching and learning on students’ learning performance: A meta-analysis and research synthesis”. *Comput. Educ*. 94, (2016) 252–275. http://dx.doi.org/10.1016/j.compedu.2015.11.008.

[5] B. Cranganu-Cretu, Z. Chen, T. Uchimoto, K. Miya, “Automatic text summarizing based on sentence extraction: A statistical approach”. *Int. J. Appl. Electromagn. Mech*. 13 (1–4), (2002) 19–23. http://dx.doi.org/10.3233/JAE-2002-513.

[6] G. Taylor, K. Barabin, K. Sayre, “An application of reinforcement learning to supervised autonomy”. (2015). https://api.semanticscholar.org/CorpusID:1203120.

[7] J. M. Sanchez-Gomez, M. A. Vega-Rodríguez, C. J. Perez, “The impact of term-weighting schemes and similarity measures on extractive multi-document text summarization”. *Expert Systems with Applications*, vol. 169, (2021) pp. 114510, 2021.

[8] G. Murray, S. Renals, “Term-Weighting for Summarization of Multi-Party Spoken Dialogues”. *In: Popescu-Belis, A., Renals, S., Bourlard, H. (eds) Machine Learning for Multimodal Interaction. MLMI 2007. Lecture Notes in Computer Science*, vol 4892. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-78155-4_14

[9] R. Khan, Y. Qian, S. Naeem, “Extractive based Text Summarization Using KMeans and TF-IDF”. *International Journal of Information Engineering and Electronic Business (IJIEEB)*, Vol.11, No.3, (2019) pp. 33-44. DOI: 10.5815/ijieeb.2019.03.05

[10] Y. SEKI, “Sentence Extraction by tf/idf and Position Weighting from Newspaper Articles”. *Proceedings of the Third NTCIR Workshop (2003)*, National Institute of Informatics.

[11] R. C. Balabantaray, D. K. Sahoo, B. Sahoo, Swain, M. “Text Summarization using Term Weights" *International Journal of Computer Applications* (0975 – 8887) Volume 38– No.1, January 2012

[12] M. Afsharizadeh, H. Ebrahimpour-Komleh, A. Bagheri, “Query-oriented text summarization using sentence extraction technique”. *in Proc. 4th Int. Conf. Web Res. (ICWR)*, Apr. 2018, pp. 128–132.

[13] R. Rani, D. K. Lobiyal, “A weighted word embedding based approach for extractive text summarization”, *Expert Systems with Applications*, Volume 186, 2021, 115867, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2021.115867.

[14] F. Douzidia, G. Lapalme, Lakhas, “an Arabic Summarising System”. *In Proceedings of the 4th Document Understanding Conferences*, (2004) pages 128–135. DUC.

[15] K. Zechner, “Fast Generation of Abstracts from General Domain Text Corpora by Extracting Relevant Sentences”. *In Proceedings of the 16th International Conference on Computational Linguistics*, (1996) 986–989, Copenhagen, Denmark.

[16] M. El-Haj, U. Kruschwitz, C. Fox, “Experimenting with Automatic Text Summarization for Arabic”. *In Zygmunt Vetulani, editor, 4th Language and Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics, LTC'09,”Lecture Notes in Artificial Intelligence*”, pages 490–499, Poznan, Poland, 2009. Springer.

[17] I. Imam, N. Nounou, A. Hamouda, H. A. Abdul Khalek, “Query Based Arabic Text Summarization”. *International Journal of Computer Science and Technology*. 4(2), 2013a, Pp. 35-39/

[18] M. Elhaj, “Multi-document Arabic Text Summarisation”. *PhD thesis, 2012, University of Essex*.

[19] M. S. Binwahlan, “Extractive Summarization Method for Arabic Text – ESMAT”. *International Journal of Computer Trends and Technology*. 21(2), (2015) pp. 103-109.

[20] M. S. Binwahlan, N. Salim, L. Suanmali, "Swarm based features selection for text summarization". *IJCSNS International Journal of Computer Science and Network Security*, 9(1), (2009b) 175–179.

[21] S. Teufel, "Deeper summarization: The second time around: An overview and some practical suggestions". *In: Lecture Notes in Computer Science (Including Sub-series Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): 9624 LNCS*, (2018) pp. 581–598. http://dx.doi.org/10.1007/978-3-319-75487-1_44.

[22] K. S. Thakkar, R. V. Dharaskar, M. B. Chandak, "Graph-based algorithms for text summarization". *In: 2010 3rd International Conference on Emerging Trends in Engineering and Technology*, (2010) pp. 516–519. http://dx.doi.org/10.1109/ICETET.2010.104.

[23] F. Amato, A. D'Acierno, F. Colace, V. Moscato, A. Penta, A. Picariello, "Semantic Summarization of News from Heterogeneous Sources. *In: Xhafa, F., Barolli, L., Amato, F. (eds) Advances on P2P, Parallel, Grid, Cloud and Internet Computing. 3PGCIC 2016. Lecture Notes on Data Engineering and Communications Technologies, vol 1. Springer, Cham*. https://doi.org/10.1007/978-3-319-49109-7_29 2017a pp. 305–314. http://dx.doi.org/10.1007/978-3-319-49109-7_29.

[24] F. Amato, V. Moscato, A. Picariello, G. Sperlí, A. D'Acierno, A. Penta, "Semantic summarization of web news". *In: Encyclopedia with Semantic Computing and Robotic Intelligence*. Vol. 01, (01), (2017b) http://dx.doi.org/10.1142/S2425038416300068, 1630006.

[25] B. Pant, V. Vimal, "Text Summarization Employing Sentence Ranking Approach". *Webology*, Volume 18, Number 5, 2021 ISSN: 1735-188X DOI: 10.29121/WEB/V18I5/52 3178 http://www.webology.org

[26] N. Andhale, L. A. Bewoor, "An overview of text summarization techniques". *In: 2016 International Conference on Computing Communication Control and Automation. ICCUBEA*, (2016) pp. 1–7. http://dx.doi.org/10.1109/ICCUBEA.2016.7860024.

[27] A. Bhola, J. Mullapudi, S. Kollipara, T. Sanaka, "Text summarization based on ranking techniques". *In: 2022 5th International Conference on Contemporary Computing and Informatics. IC3I*, pp. 1463–1467. http://dx.doi.org/10.1109/IC3I56241.2022.10072962.

[28] P. Verma, A. Verma, P. Sukomal, "An approach for extractive text summarization using fuzzy evolutionary and clustering algorithms". *Applied Soft Computing* 120 (2022) 108670

[29] N. Alami, M. E. Mallahi, H. Amakdouf, H. Qjidaa, "Hybrid method for text summarization based on statistical and semantic treatment". *Multimedia Tools Appl*. 80 (13), (2021) 19567–19600. http://dx.doi.org/10.1007/s11042-021-10613-9.

[30] S. Singh, J. P. Singh, A. Deepak, "Supervised weight learning-based PSO framework for single document extractive summarization". *Applied Soft Computing Journal* 161 (2024) 111678

[31] E. Hovy, C. Y. Lin, "Automatic text summarization in SUMMARIST". *In Proceedings of the ACL'97/EACL'97 workshop on intelligent scalable text summarization* (pp. 18–24), Madrid, Spain.

[32] Y. Gong, X. Liu, "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 (SIGIR'01)* (pp. 19–25), New Orleans, LA, USA.

[33] S. P. Pati, R. Rautray, "Sentence Selection for Extractive Text Summarization using TOPSIS Approach". *Procedia Computer Science* 235 (2024) 1532–1538

[34] J. Kupiec, J. Pedersen, F. Chen, "A trainable document summarizer". *In Proceedings of the ACM. SIGIR conference*. July 1995. New York, USA, 68-73.

[35] C. Y. Lin, E. Hovy, "Identifying topics by position". *In Proceedings of the Fifth conference on Applied natural language processing*. March. San Francisco, CA, USA, 283-290, 1997.

[36] C. Y. Lin, "Training a selection function for extraction". *In Proceedings of the Eighteenth Annual International ACM Conference on Information and Knowledge Management (CIKM)*. 2-6 Nov. 1999. Kansas City, Kansas, 55-62.

[37] J. M. Conroy, D. P. O'leary, "Text summarization via hidden markov models". *Proceedings of SIGIR '01*. 9-12 September 2001. New Orleans, Louisiana, USA, 406-407.

[38] M. Osborne, "Using maximum entropy for sentence extraction". *Proceedings of the ACL'02 Workshop on Automatic Summarization*. July 2002. Morristown, NJ, USA, 1-8.

[39] K. Svore, L. Vanderwende, C. Burges, "Enhancing single document summarization by combining RankNet and third-party sources". *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. June 2007. Prague: Association for Computational Linguistics, 448–457.

[40] M. A. Fattah, F. Ren, "GA, MR, FFNN, PNN and GMM based models for automatic text summarization". *Computer Speech and Language*. 2008. 23(1), 126-144.

[41] W. Wang, S. Li, J. Li, W. Li, F. Wei, "Exploring hypergraph-based semi-supervised ranking for query-oriented summarization", *Information Sciences* 237 (2013) 271–286.

[42] H. K., Gianey, R. Choudhary, "Comprehensive review on supervised machine learning algorithms". *In: Proceedings - 2017 International Conference on Machine Learning and Data Science*. MLDS 2017 2018-Janua, pp. 38–43. http://dx.doi.org/10.1109/MLDS.2017.11.

[43] J. Thomas, A. Sreeraj, A. Sreeraj, M. M. Varghese, T. Kuriakose, "Automatic Text Summarization Using Deep Learning and Reinforcement Learning". (2022) pp. 769–778. http://dx.doi.org/10.1007/978-981-16-5157-1_60.

[44] S. Kumar, "Machine learning (supervised)". *In: International Series in Operations Research and Management Science*. Vol. 264, (2019) pp. 507–568. http://dx.doi.org/10.1007/978-3-319-68837-4_16.

[45] H. Dalianis, M. Hassel, J. Wedekind, D. Haltrup, K. de Smedt, & T. L. Christopher, "From SweSum to ScandSum- Automatic text summarization for the Scandinavian languages". *In Holmboe, H. (ed.) Nordisk Sprogteknologi 2002: Årbog for Nordisk Språkteknologisk Forskningsprogram*, 2000-2004 (pp. 153-163). Copenhagen: Museum Tusculanums Forlag

[46] P. R. Dedhia, H. P. Pachgade, A.P. Malani, N. Raul, M. Naik, "Study on abstractive text summarization techniques". *In: 2020 International Conference on Emerging Trends in Information Technology and Engineering. Ic-ETITE*, pp. 1–8. http://dx.doi.org/10.1109/ic-ETITE47903.2020.087.

[47] P. Verma, D. P. Tyagi, "Supervised Machine Learning: Predispositions, Practices and Perspectives", (2020), https://api.semanticscholar.org/CorpusID:233228017.

[48] F. Wu, T. Zheng, L. Yao, H. Feng, "A new unsupervised Algorithm for extracting relationship words between two entities". *In: 2021 3rd International Conference on Advances in Computer Technology, Information Science and Communication, CTISC*, pp. 161–165. http://dx.doi.org/10.1109/CTISC52352.2021.00037.

[49] L. Martin, "Automatic Sentence Simplification Using Controllable and Unsupervised Methods". *Computation and Language, Sorbonne Université, (Issue 2021SORUS265)*. https://theses.hal.science/tel-03543971.

[50] J. Cheng, F. Zhang, X. Guo, "A syntax-augmented and headline-aware neural text summarization method". *IEEE Access 8*, 218360–218371(2020), http://dx.doi.org/10.1109/ACCESS.2020.3042886.

[51] V. K. Verma, A. Yadav, T. Jain, "Key Feature Extraction and Machine Learning-Based Automatic Text Summarization". *In: Abraham, A., Dutta, P., Mandal, J., Bhattacharya, A., Dutta, S. (eds) Emerging Technologies in Data Mining and Information Security. Advances in Intelligent Systems and Computing*, (2019), vol 814. Springer, Singapore. https://doi.org/10.1007/978-981-13-1501-5_76

[52] K. Thirumoorthy, J. J. J. Britto, "A hybrid approach for text summarization using social mimic optimization algorithm". *Iranian J. Sci. Technol., Trans. Electr. Eng*. 47 (2), (2023) 677–693. http://dx.doi.org/10.1007/s40998-022-00572-8.

[53] M. H. H. Wahab, N. A. W. A. Hamid, S. Subramaniam, R. Latip, M. Othman, "A Decomposition–based multi-objective differential evolution for extractive multi-document automatic text summarization". *Applied Soft Computing Journal* 151 (2024) 110994

[54] EASC," Essex Arabic Summaries Corpus", [Online] Available: http//:privatewww.essex.ac.uk/~melhaj/easc.htm, (14-01-2013.)

[55] C. Y. Lin, "ROUGE: A Package for Automatic Evaluation of Summaries". *Proceedings of the Workshop on Text Summarization Branches Out, 42nd Annual Meeting of the Association for Computational Linguistics*. 25–26 July (2004), Barcelona, Spain, 74-81.

[56] H. Van Lierde, T. W. S. Chow, "Learning with fuzzy hypergraphs: A topical approach to query-oriented text summarization". *Information Sciences* 496 (2019) 212-224, https://doi.org/10.1016/j.ins.2019.05.020

[57] G. Salton, A. Wong S. Yang, "A Vector Space Model for Automatic Indexing". *Communications of the ACM*, vol. 18, no. 11, (1975) pp. 613–620.

[58] M. Peng, B. Gao, J. Zhu, J. Huang, M. Yuan, F. Li, "High quality information extraction and query-oriented summarization for automatic query-reply in social network". *Expert Systems with Applications* 44 (2016) 92–101

[59] M. Yousefi-Azar, L. Hamey, "Text summarization using unsupervised deep learning". *Expert Systems with Applications*, 68, (2017) 93–105. https://doi.org/10.1016/j.eswa.2016.10.017

[60] F. Kiyani, O. Tas, "A survey automatic text summarization". *Pressacademia*, 5(1), (2017) 205–213. https://doi.org/10.17261/pressacademia.2017.591

[61] Abdi, S. M. Shamsuddin, R. M. Aliguliyev, "QMOS: Query-based multi-documents opinion-oriented summarization". *Information Processing and Management* 54 (2018) 318–338.

[62] F. Geng, Q. Liu, P. Zhang, "A time-aware query-focused summarization of an evolving microblogging stream via sentence extraction". *Digital Communications and Networks* 6 (2020) 389 – 397.

[63] M. Alhoshan, N. Altwaijry, AUSS: "An Arabic query-based update-summarization system". *Journal of King Saud University – Computer and Information Sciences* 34 (2022) 3732–3743.

[64] R. M. Aliguliyev, "Clustering techniques and discrete particle swarm optimization algorithm for multi-document summarization". *Comput. Intell.* 26 (4) (2010) 420–448.

[65] R. M. Alguliyev, R. M. Aliguliyev, N. R. Isazade, "An unsupervised approach to generating generic summaries of documents". *Applied Soft Computing* 34 (2015) 236–250

[66] N. Salim, "Analysis and Comparison of Molecular Similarity Measures". *University of Sheffield: Ph. D. Thesis*, (2002).

[67] S. Khoja, R. Garside, "Stemming Arabic text. Computing Department", *Lancaster University, Lancaster,* (1999), <http://www.comp.lancs.ac.uk/computing/users/khoja/stemmer.ps>

[68] G. Salton, C. Buckley, "Term-weighting approaches in automatic text retrieval". *Information Processing Management*, 24 (5) (1988) 513–523, http://dx.doi.org/10.1016/0306-4573(88)90021-0.

[69] I. Mani, "Automatic Summarization". *(1st ed.) (2001), Amsterdam: John Benjamins Publishing Company*.