Benha Journal of Applied Sciences (BJAS) **print: ISSN 2356–9751**
Vol. (9) Issue (*) (2024), (17-27) **online: ISSN 2356–976x**
http://bjas.bu.edu.eg

An Integrated System for Accessible Summarization of Web Search Results for the Blind and Visually Impaired

Mai A. Bakeer, Khaled M. Fouad and Tarek EL-Shishtawy
Faculty of Computers and Artificial Intelligence, Benha University, Benha, 13511, Egypt
E-mail: mai.ashraf@fci.bu.edu.eg

Abstract

Blind and visually impaired individuals encounter significant challenges when accessing search results online, primarily due to the vast amount of information available and the lack of adequate support tools tailored to their needs. While existing solutions like screen readers facilitate sequential exploration of search results and some tools organize results based on user behavior, there remains a gap in providing efficient online summaries of queries specifically tailored for this demographic. This study proposes an integrated system designed to address this gap by providing summarized search results, with a focus on accommodating the Arabic language. Two distinct approaches to online query-based summarization were introduced. The first approach aims to identify the most relevant sentence to the query from the search results, while the second approach utilizes an adaptive technique to extract the most pertinent sentences from the first 'n' documents in the search results. Comparative evaluations were conducted, with the first approach being benchmarked against Google Assistant, while the second approach was assessed based on summaries provided by two human experts and GPT-4. Results indicate that the first approach generally outperformed Google Assistant. Furthermore, experimental evaluations demonstrated the close alignment of the results retrieved by the second approach with the summaries provided by GPT-4, with an average relevancy score of 0.92, and the highest similarity scores of the second proposed system with Experts 2 and 1 are 0.93 and 0.85, respectively. These findings underscore the utility of the proposed system in facilitating access to information for blind and visually impaired individuals.

Keywords *Online summarization, Extractive multi-document summarization, Blind, Visually Impaired, Text-to-Speech*

1. Introduction

Worldwide, the International Agency for the Prevention of Blindness (IAPB) reported in 2020 that there are 43 million people who are blind, 295 million experiencing moderate to severe vision loss, and within North Africa and the Middle East, there are 3.1 million individuals who are blind. [1]. Individuals with blindness or visual impairments encounter considerable obstacles when navigating online information, particularly with search engine outputs. This is due to the overwhelming volume of data and the absence of specialized assistance tools. In contrast to those with sight, they cannot swiftly skim through documents to gauge their significance. Instead, they invest additional time and effort utilizing screen readers or other tools that sort search results according to user activity to comprehend the documents' relevance. [2] [3] Research on assistive technologies for the blind and visually impaired (VI) has focused on two main areas. Firstly, efforts have been made to condense documents like books or news into more manageable forms. For instance, Shakila [4] employed a technique to distill books into keywords using weighted TFIDF and Part-Of-Speech (POS) tagging, while Xiaojun [5] developed a system named Braillesum that leverages an Integer Linear Programming(ILP)-based framework to create improved Braille summaries of news articles. Secondly, initiatives have aimed to streamline web browsing. Aboubakr [6], [7] worked on structuring search results into concepts to aid VI individuals in more effectively navigating these results. However, feedback from evaluation sessions indicated a preference for a per-webpage summary to serve as a guide, helping users decide whether to engage with the current page or seek another that better aligns with their research needs. Najd [8] created a proxy service to ease web browsing by customizing webpages to user preferences. Nonetheless, gaps remain in [6] and [7] as they only provide concepts not a summary, in [8] since webpages are different in their layout, processing a webpage takes time, and does not provide a summary, and in [4], [5] they provide a summary of only one book or news article and take time to provide the summary. So, there's still a gap in providing efficient online summaries specifically designed for this demographic. This study aims to bridge this gap by proposing an integrated system tailored for blind and VI individuals, focusing on summarizing Google search results.

The organization of this paper is outlined as follows: Section **two** covers the existing research on summarization systems specifically for the blind and visually impaired, as well as broader multi-document summarization strategies. Section **three** details the tools and methodologies employed in this study. Section **four** describes the two methods being introduced. Following that, Section **five** presents the findings and discussions, including the assessment of the system. The paper concludes with Section **six**, which provides final thoughts and potential directions for future research.

2. Literature Review

Navigating web searches can be a challenging endeavor, and it's even more so for individuals who are blind or have visual impairments. Therefore, the ability to explore and condense search results is crucial in assisting these individuals to perform web searches with ease. Numerous studies have been conducted to address the challenge of automatic text summarization. These studies employ a variety of techniques and algorithms, which are typically classified according to their intended use—be it generic or query-specific—and the type of input they handle, whether it's a single document or multiple documents. Generic summaries aim to encapsulate the entire scope of relevant topics within the original text, whereas query-focused summaries concentrate on topics or keywords specified by the user. Summarization of a single document involves distilling key information from that document alone, while multi-document summarization involves extracting pertinent information across several documents. [9], [10]

The subsequent sections delve into the various methods for summarizing text documents, as well as the strategies employed to navigate and condense search results specifically for the benefit of blind and visually impaired individuals.

2.1 Generic-based summarization

First, the Generic-based single-document summarization approaches are mentioned as shown below.

Dima Suleiman et al. [11] introduced a model for abstracting Arabic texts using sequence-to-sequence recurrent neural networks. This model featured a multi-tiered encoder and a singular decoder. The encoder utilized bidirectional long-short-term memory (LSTM) networks, while the decoder operated on unidirectional long-short-term memory networks. The encoder was composed of layers for input text, keywords, and named entities. The decoder incorporated a global attention mechanism that considered all hidden states of the input to produce words for the summary. The experiment's dataset was specially gathered for the purpose of abstract summarization of Arabic texts from various sources. The model's performance was assessed through both quantitative and qualitative metrics. Beyond ROUGE1, three additional scales were introduced—ROUGE1-NOORDER, ROUGE1-STEM, and ROUGE1CONTEXT—to better capture the essence of abstract summaries and the nuances of the Arabic language.

Dimah Alahmadi et al. [12] introduced a novel method known as the topic-aware abstractive Arabic summarization model (TAAM), which utilizes a recurrent neural network (RNN) to enhance the accuracy of abstractive summarization in Arabic. This is in response to the scarcity and syntactical issues of existing Arabic summarization models. The TAAM approach comprises four key components: 1) A word-embedding module that transforms the input text into multi-dimensional vectors. 2) An encoder module with several RNN layers equipped with LSTM gates, designed to compute the hidden state while prioritizing significant words from the input documents. 3) A topic-aware module that guides the model to align with human-like understanding. 4) A decoder module that assesses the likelihood of each word, selecting the most suitable sentences for the summary. The model's effectiveness was gauged through two evaluation methods: a quantitative assessment using the ROUGE metric and a qualitative review conducted by human volunteers.

Second, the Generic-based multi-document summarization approaches are mentioned as shown below.

Praveen K. Wilson et al. [13] developed the Spider Monkey Optimization (SMO) method to generate summaries from several documents, aiming to enhance the speed of execution. The method utilized two types of features—grammatical and semantic—to identify pertinent sentences for summarization. The process involved condensing multiple documents into one, followed by a preprocessing stage to refine the content. Subsequently, the document's grammatical and semantic features were extracted. The SMO technique was then applied to create the summary. For experimental validation, datasets from BBC news, DUC 2002, 2006, and DUC 2007 were employed, with evaluation metrics including recall, F-measure, and precision.

Rebhi S. Baraka et al. [14] proposed an approach for auto-summarization of extensive Arabic texts from numerous documents by employing genetic algorithms alongside the MapReduce parallel programming framework. Key text attributes, including readability and cohesion, were extracted during the feature extraction phase. These attributes were then utilized to assess sentences within the genetic algorithm process. The genetic algorithm was integrated with MapReduce tasks, structuring each iteration of the genetic algorithm as an individual MapReduce job.

2.2 Query-Based Summarization

First, the Query-based single-document summarization approaches are mentioned as shown below.

Rasha M. Badry et al. [15] introduced a model for creating concise summaries from extensive Arabic documents, tailored to specific queries. This model harnessed the Latent Semantic Analysis (LSA) strategy and the Arabic WordNet (AWN) ontology. It featured a five-part framework: enlarging the query scope, initial processing, generating the input matrix, executing Singular Value Decomposition (SVD) and its reduced variant, and finally, evaluating and prioritizing sentences. The algorithm's effectiveness depended on the degree of resemblance between the user's queries, the augmented queries, and the sentences within the original text.

Second, the Query-based multi-document summarization approaches are mentioned as shown below.

Jesus M. Sanchez-Gomez et al. [16] developed a multi-objective optimization strategy to enhance query-based multi-document text summarization, focusing on improving query relevance and minimizing redundancy. The Multi-Objective Shuffled Frog-Leaping Algorithm (MOSFLA) was crafted, executed, and refined to address these objectives. As a collective swarm intelligence algorithm, MOSFLA integrated novel operators tailored for these specific goals and was optimized for multi-objective challenges. The model's performance was tested using data from the Text Analysis Conference (TAC), particularly TAC2009, and evaluated using metrics such as ROUGE-1, ROUGE-2, and ROUGE-SU4.

Nabil Alami et al. [17] crafted a summarization method for Arabic texts that utilizes a variational auto-encoder (VAE) to distill a feature space from voluminous input data. The VAE was employed to derive a condensed conceptual space from the large-scale input. Their VAE's intricate structure was bifurcated: initially, the encoder transformed sentences from the term vector domain to the latent semantic domain. Subsequently, the decoder undertook unsupervised data reconstruction to estimate significance by reconstituting the latent semantic domain and the observed term vectors. The authors acknowledged two primary drawbacks of their method: the protracted training duration for substantial datasets and the complexity of pinpointing the network's optimal parameters. Despite this, they conducted exhaustive experiments to identify the most effective parameters, which proved to be a time-intensive endeavor.

2.3 Text Summarization for Blind and VI People

N. A. Al-Mouh et al. [8] acknowledged the obstacles VI individuals encounter while navigating and extracting information from the Internet using conventional methods. To address this, they introduced a proxy service that can be integrated to alleviate the browsing issues faced by VI individuals and enhance their web experience. This VI People-aware proxy service is designed to ease web navigation on any device for VI users by providing them with a customized webpage featuring content that is condensed and tailored to their pre-set preferences. However, they encountered several challenges: visual presentations can be ambiguous, making it difficult to discern the boundaries of a segment; webpages often have varying layouts and may not be neatly organized or properly labeled; processing a webpage is time-consuming; dynamic webpages, JavaScript, and Web 2.0 elements can complicate the process; and the dynamic nature of web content poses additional hurdles. These challenges have impacted the proxy's performance, leading to the consideration of each website as a unique case.

Xiaojun Wan et al. [5] addressed the summarization difficulties faced by blind and VI individuals who are unable to read standard news documents as easily as those with sight, relying instead on braille or specialized devices. To improve this situation, they introduced a system named BrailleSUM, designed to generate superior braille summaries. This system integrates the braille length of sentences from news articles into a well-known ILP-based summarization framework. Tests conducted on a DUC dataset and assessed using the ROUGE metric revealed that BrailleSUM can create braille summaries that are more concise than those produced by previous methods without compromising the quality of the content.

Aboubakr Aqle et al. [7] discovered that VI users often spend considerable time and effort navigating through numerous search results with screen readers to find their desired content. To streamline this process, they developed a model that restructures search engine outcomes specifically for VI users. The outcome was a working prototype that condensed search results into key concepts. Through iterative testing, the prototype was refined until it consistently delivered results that aligned with the VI users' search intentions. Feedback from evaluation sessions indicated a preference among users for a webpage summary feature, serving as a gateway for deciding whether to engage with the current page or seek an alternative that better matches their research needs.

Shakila Basheer et al. [4] employed a methodical approach to distill books into keywords by utilizing the weighted TFIDF, which streamlines the process by eliminating the need to read the entire text repeatedly. The authors implemented a POS tagging feature to classify terms, ensuring the exclusion of irrelevant phrases. In their analysis, they focused solely on NOUNS and VERBS, as these parts of speech were more effective in generating descriptive content.

Aboubakr Aqle et al. [6] introduced an innovative search engine interface, named InteractSE, designed to assist VI individuals in navigating search results with greater ease. The interface groups search outcomes into concepts and arranges them in a multi-tiered tree structure reflecting their hierarchical relationship. The extraction and clustering of these concepts are performed through the Formal Concept Analysis (FCA) method, a mathematical model for analyzing data. To assess its effectiveness, InteractSE was tested against the traditional Google search interface by a group of 16 blind users.

3. Methods

The study utilized a range of principal techniques and instruments specifically designed to streamline the summarization task for individuals who are blind or VI. These elements are central to the study's approach to enhancing accessibility in text summarization:

3.1 Speech Recognition*:* is a component of Natural Language Processing (NLP). It simply refers to the capability of software to identify spoken words and phrases and transcribe them into human-readable text. This technology is utilized in various applications, including voice assistant systems, voice-activated chatbots, and interactive voice response robots, among others [18].

The voice query is converted into text through the Google Speech Recognition API available in the SpeechRecognition Python Library, which conveniently doesn't necessitate any API keys [19]. This simplifies the search process for individuals who are blind or have visual impairments.

3.2 Wiki Search: Wikipedia serves as a reservoir of information. The Wikipedia Python library is utilized for its ease in retrieving and interpreting data, conducting searches, and extracting elements such as content, links, and images from pages. However, the library's search function may not always yield precise results. To enhance accuracy, Google search is employed to obtain the URLs of relevant Wikipedia pages. Additionally, a bespoke script is used to navigate Wikipedia pages, leveraging the requests library to fetch the URL and retrieve the page's HTML content, and the bs4 library to parse the HTML and isolate the page title. Subsequently, other methods from the Wikipedia library are applied to access the pages identified in the search results and extract the desired data from them.
[20].

3.3 Data Pre-processing: Natural language represents our primary mode of expression and communication. The convergence of fields like artificial intelligence, linguistics, formal languages, and computer science has led to the emergence of an interdisciplinary field referred to as NLP [21].

Basic techniques from the field of NLP are utilized on both the user's query and the documents to normalize them, simplifying the processing. The subsequent pre-processing procedures are executed using the Natural Language Toolkit (NLTK) Python library:

3.3.1 Sentence tokenization involves segmenting input documents into individual sentences, a process performed using the sentence tokenizer from the NLTK library. [22].

3.3.2 Word tokenization refers to the process of breaking down sentences into their constituent words, which is carried out using the word tokenizer provided by the NLTK library. [22].

3.3.3 Stop words removal Eliminating stop words involves removing frequently occurring words that aid in sentence structure but do not significantly alter the sentence's meaning, such as "على" (on), "بعد" (after), and "في" (in) [21]. This is done by employing the Arabic stop words list from the NLTK to filter out these high-frequency, non-essential words [22].

3.3.4 Stemming is a method that truncates the endings of words to find their root form. For instance, "حرك" (mov) serves as the foundational stem for various related words such as "تحريك" (moving), "حركات" (movements), and "يتحرك" (moves). The ISRI Arabic Stemmer within the NLTK is utilized for its efficiency in reducing Arabic words to their stem forms, ensuring a swift processing time. [23].

3.4 Text Similarity: involves assessing the semantic resemblance between pairs of sentences, such as the query and document sentences, or among the sentences themselves. This is done using the Jaccard index, a statistical tool for gauging the similarity and diversity of sample sets. The Jaccard index is calculated by dividing the magnitude of the intersection by the magnitude of the union of the sample sets.
[24], [25].

3.5 SPEECH SYNTHESIS: involves the steps of analyzing, processing, comprehending, and ultimately transforming written text into speech that sounds natural and intelligible [18]. This function is performed using the OpenAI Text-to-Speech library [26]. The choice of this library is due to the high quality of its output, which closely resembles human speech, thus providing a satisfactory experience for blind and visually impaired users.

4. Proposed System

The system we've designed adopts a comprehensive method to condense web search findings for blind and VI users, emphasizing the Arabic language. We introduce two innovative methods for real-time query-driven summarization: the initial method selects the sentence that most closely aligns with the query from the top document in the search results, and the second method identifies the closest matching sentences from the top 'n' documents through a flexible approach. As shown in Figure 1, our system framework integrates speech recognition, linguistic preprocessing, summarization, and speech synthesis modules to ensure a smooth and user-friendly experience.

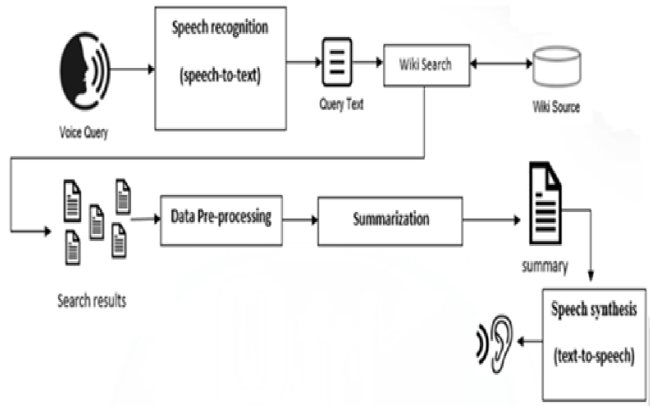

Fig. (1) The proposed integrated system for summarizing search engine results for Blind and VI individuals.

4.1 First approach: The initial method draws inspiration from the Google snippet, a concise excerpt of text displayed below a website's link in Google's search outcomes, serving as a condensed overview of the search result [27]. This technique relies on identifying the sentence from the foremost document in the search results that exhibits the highest similarity to the query, determined through the Jaccard technique. The structure of this method is illustrated in Figure 2.

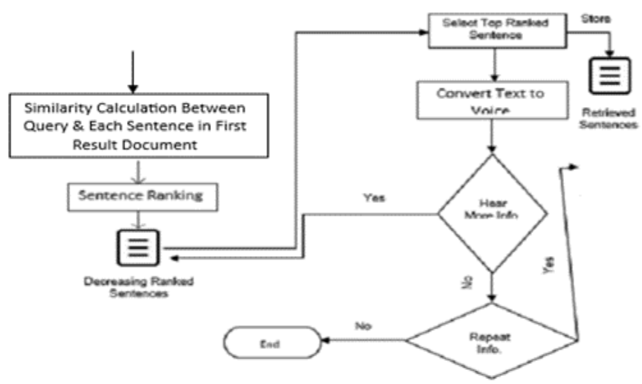

Fig. (2) The architecture of the first approach

4.1.1 Similarity Calculation:

The Jaccard technique is employed to determine the similarity by comparing the root forms of the query with those of each sentence. Consequently, every sentence is assigned a similarity score in relation to the query:

$$\mathbf{SSQ_i} = \frac{\mathbf{TQ \cap Sent_i}}{\mathbf{TQ \cup Sent_i}} \qquad \mathbf{(1)}$$

Where:

- (SSQ_i) represents the similarity score of the sentence with the query, where ($i = 1, 2, \ldots\ldots, n$) sentences in the document.
- (TQ) denotes the text of the query.
- ($TQ \cap Sent_i$) signifies the cardinality of the intersection between the stems of the query and the stems of sentence (i).
- ($TQ \cup Sent_i$) represents the cardinality of the union between the stems of the query and the stems of sentence (i).

Input: Voic Query (VQ)
Output: Voice of Summary List (V(SL))
1: Convert VQ to TQ Text Query
2: Search for the Text Query on Wikipedia using Google search library, then select the first result document
3: Apply pre-processing steps on the Text Query
4: Apply sentence tokenization to the document (D)
5: For each sentence ($sent_i$) in D
  i: Apply word tokenization to $sent_i$
  ii: For each term t_j in $sent_i$
    a: Apply stem rules to t_j
  iii: Calculate sentence similarity score with the query (SSQ_i) =Intersection_Card / Union_Card
  iv: Add SSQ_i to list of sentence score (L(SS))
6: Descending order L(SS)
7: Select sentence with highest score (SHS), then add SHS to SL
8: Convert SHS to voice
9: Repeat from step 7 for more info.
10: Convert SL to voice
11: return V(SL)

Fig. (3) The Algorithm of the fist approach

4.2 Second approach:

This method focuses on extracting sentences that closely match the query from the initial ‘n’ documents in the search results, utilizing an adaptive technique. The design of this second approach is depicted in Figure 4.

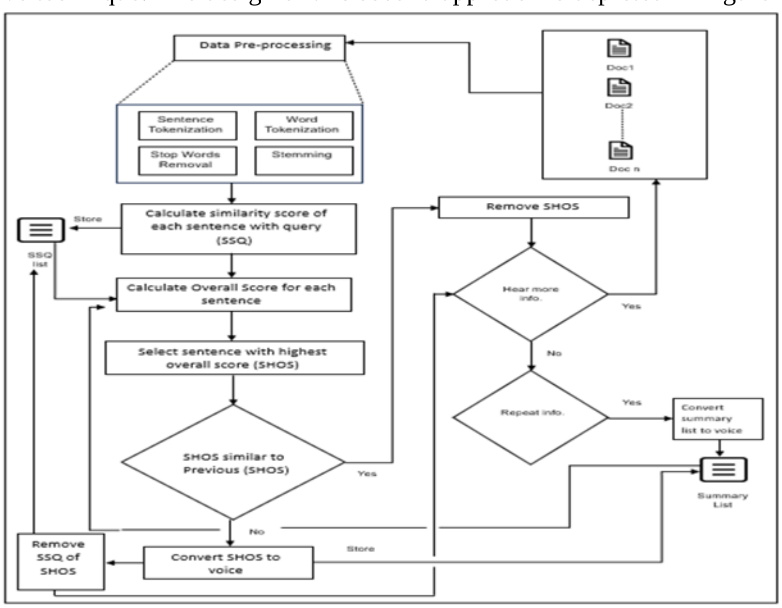

Fig. (4) The architecture of the second approach

4.2.1 Features of Our Summarization Technique in the Second Approach:

Our approach evaluates the resemblance of text at both the single document level and collectively across several documents. It considers how pertinent sentences are to the query and maintains a variety in sentence choice to avoid redundancy in the summarization. The method is flexible (adaptive) and operates in real-time, facilitating swift and straightforward summarization that retains the essence of each document's context.

4.2.2 Similarity Calculation:

An adaptive approach has been developed to calculate the relevance between the query and sentences:

Jaccard Similarity Score: The Jaccard similarity method is used to compute the similarity score. This score for each potential sentence is assessed by considering the shared and distinct terms in both the query and the sentence.

$$SSQ_j = \frac{Intersection\_cardinality}{Union\_Cardinality} \quad (1)$$

Where:

- (SSQ_j) is the sentence similarity score with query, (j = 1,….., n) sentences in Document (D_i), (i=1,….,m)
- (Intersection_cardinality = $TQ \cap Sent_j$) represents the cardinality of the intersection between the query and sentence (j)
- (Union_Cardinality = $TQ \cup Sent_j$) denotes the cardinality of the union between the query and sentence (j).
- ($SSQ_j \neq 0$).

$$L(SSQ) += SSQ_j \quad (2)$$

Wher

- (L(SSQ)) is a list of sentence similarity scores with the query.

e:

Overall score for each sentence: The total score for each sentence is determined by considering both the similarity score in relation to the query and the score of the sentence selected prior to it.

.

$$OS_i(s)_{ln} = \sum_{x=1}^{ln-1} \frac{SSQ_i}{SP_x} \quad (3)$$

Where:

- ($OS_i(s)$) is the overall score of sentence (i), (i = 1,……,len(L(SSQ))).
- (SP_x) represents the score of the previous selected sentence (the maximum score of (OS) list in level (x) number).

$$SP_x = \begin{cases} 1 & if\ ln = 1, else \\ Max(L(OS) & in\ x\ ln \end{cases} \quad (4)$$

Where:

- (ln) is the level number and document number.

$$L(OS) += OS_i(s)_{ln} \quad (5)$$

Where:

- L(OS) is the list of overall scores of sentences in level number (ln).

Selection of the Sentence with the Highest Overall Score: The sentence that achieves the top overall score is chosen to be part of the summary.

$$HOS = Max(L(OS)_{ln} \quad (6)$$

Where:

- HOS is the highest overall score in level number (ln).

Similarity Score between the Selected Sentence and Previous Selected Sentences: The similarity score is determined by comparing the chosen sentence with the previously selected sentences.

$$SSPS_x = \frac{Intersection\_cardinality}{Union\_cardinality} \quad (7)$$

Where:

- (SSPS) is the similarity score between the sentence with the highest overall score at level number (ln) and previous selected sentence at level number (x), (x=1,….,ln-1).
- (Intersection_cardinality = $SHOS \cap PSS_x$).
- (Union_cardinality = $SHOS \cup PSS_x$).
- (SHOS) is the sentence with the highest overall score in level number (ln).
- (PSS_x) is the previous selected sentence at level number (x).

If the chosen sentence bears resemblance to any previously chosen sentences, it is omitted. If not, it is incorporated into the summary using **Eq 8** and eliminated from the sentence pool for calculating similarity scores with the query as per **Eq 9**.

$$SL += SHOS \quad (8)$$
$$L(SSQ) -= SSQ_{HOS} \quad (9)$$

The process is reiterated as necessary to gather additional information. The adaptive summarization method is depicted in Figure 5.

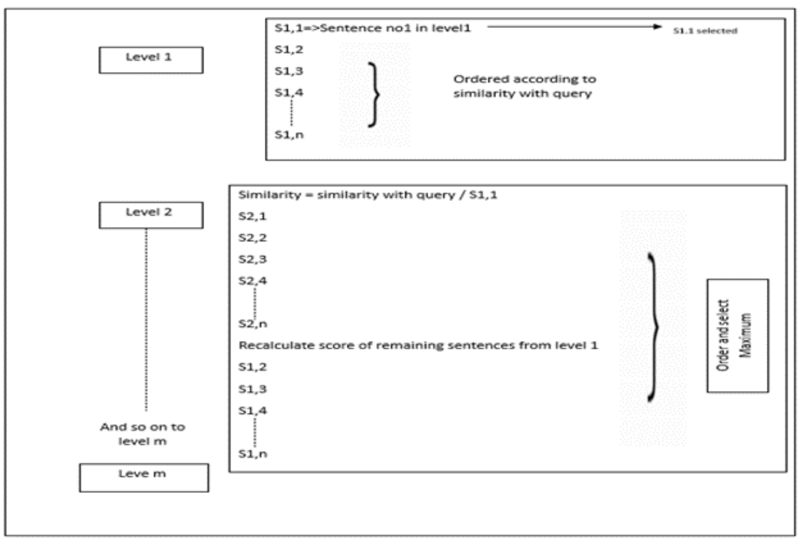

Fig. (5) Simple architecture for the adaptive summarization technique

Input: Voice Query (VQ)
Output: Voice of Summary List (V(SL)) from first top n results from Arabic Wikipedia
1: Convert VQ to TQ Text Query
2: Search for the Text Query on Wikipedia using Google search library
3: Apply pre-processing steps on the Text Query
4: For each Document D_i in n results
    a: Apply sentence tokenization to D_i
    b: For each sentence $Sent_j$ in D_i
        i: Apply word tokenization to $Sent_j$
        ii: For each term t_q in $sent_j$
            1: Apply stem rules to t_q
        iii: Calculate sentence similarity score with query (SSQ_j) = Intersection_Card/ Union_Card,
            $SSQ_j \neq 0$
        iv: Add SSQ_j to list of sentence similarity score with query (L(SSQ))
    C: Calculate overall score of sentence z ($OS_z(s)_{ln}$) = $\sum_{x=1}^{ln-1} \frac{SSQ_i}{SP_x}$
        i: If ln == 1 then
            1: $Sp_x = 1$
        ii: Else
            1: Sp_x = max(L(OS)) in x ln
    d: Add $OS_z(s)_{ln}$ to list that we chose from it the highest OS → L(OS)
    e: Select sentence with highest overall score in level number ln (HOS) = $Max(L(OS))_{ln}$
    f: Calculate similarity score between sentence with the highest overall score (SHOS) and previous selected sentences (PSS)
    g: for each PSS_x in PSS
        i: $SSPS_x = \frac{SHOS \cap PSS_x}{SHOS \cup PSS_x}$
        ii: If $SSPS_x$ == 1
            1: Remove this sentence SHOS
            2: Break
    h: If $SSPS_x$!= 1
        i: Convert SHOS to voice
        ii: Add SHOS to SL List
        iii: Remove SSQ_{HOS} from L(SSQ)
5: Convert SL to voice to repeat the heard information while preserving the context of each sentence

Fig. (6) The Algorithm of the second approach

5. Results and Discussion

We conducted experimental evaluations to assess the performance of our proposed system against established benchmarks and human expert summaries. Our results demonstrate that both approaches significantly outperform existing tools, such as Google Assistant, in terms of relevance and accuracy. Furthermore, our system achieved high levels of similarity to Generative Pre-Trained Transformer (GPT-4) [28] and human expert summaries, indicating its effectiveness in providing accessible and relevant summaries of web search results for blind and VI users.

A human expert evaluated the performance of the initial proposed system by comparing its results with those from Google Assistant. It was observed that the first proposed system's results were superior to Google Assistant's in **60%** of the cases, and Google Assistant provided no results in **15%** of the instances.

The evaluation of the second proposed system's effectiveness reveals that its summaries are of high relevance and bear a strong resemblance to those produced by GPT-4. **Table 1** indicates that the highest and lowest similarity percentages of the second proposed system with GPT-4 are **96%, 90%** respectively and the average relevancy percentage is **92%**.

TABLE (1) ***Results of the performance evaluation of the second proposed system according to GPT-4***

	GPT-4
Maximum Similarity with the 2nd system	0.96
Minimum Similarity with the 2nd system	0.90
Average Relevancy Score	0.92

The evaluation of the second proposed system's effectiveness reveals that its summaries are of high relevance and bear a strong resemblance to those produced by human experts. **Table 2** indicates that the highest and lowest similarity percentages of the second proposed system with expert 1 and expert 2 are **85%, 62%** and **93%, 68%**, respectively and the average percentage is **79%** for both expert 1 and expert 2.

TABLE (2) ***Results of the performance evaluation of the second proposed system according to human expert summaries 1 and 2***

	Expert 1	Expert 2
Maximum Similarity with the 2nd system	0.85	0.93
Minimum Similarity with the 2nd system	0.62	0.68
Average Relevancy Score	0.79	0.79

6. Conclusion and Future work

In conclusion, we have presented an integrated system designed to enhance the accessibility and usability of web search results for blind and VI individuals. Our proposed system offers accessible summarization techniques tailored to the unique needs of this user demographic, significantly improving their ability to navigate and comprehend online information. Future research directions may involve expanding the system's capabilities to support additional languages and information sources, further enhancing its utility and accessibility for blind and VI users worldwide.

References

[1] . R. Bourne, J. Steinmetz, S. Flaxman and et al, "Trends in prevalence of blindness and distance and near vision impairment over 30 years: an analysis for the Global Burden of Disease Study," *Lancet Glob Health,* vol. 9, no. 2, pp. 130-143, 2021.

[2] S. Naseer, U. Rashid, . M. Saddal, A. R. Khan, . Q. Abbas and Y. Daadaa, "WSREB Mechanism: Web Search Results Exploration Mechanism for Blind Users.," *Applied Sciences,* vol. 13, no. 19, p. 11007, 2023.

[3] A. Alluqmani, M. A. Harvey and Z. Zhang, "The Barriers to Online Clothing Websites for Visually Impaired People: An Interview and Observation Approach to Understanding Needs," in *Proceedings of the 2023 ACM Designing Interactive Systems Conference*, New York, 2023.

[4] S. Basheer, M. Anbarasi, D. G. Sakshi and V. V. Kumar, "Efficient text summarization method for blind people using text mining techniques," *International Journal of Speech Technology,* pp. 1-13, 2020.

[5] X. Wan and Y. Hu, "Braillesum: A news summarization system for the blind and visually impaired people," *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing,* vol. ‖, pp. 578-582, 2015.

[6] A. Aqle, D. Al-Thani and A. Jaoua, "Can search result summaries enhance the web search efficiency and experiences of the visually impaired users?," *Universal Access in the Information Society ,* vol. 1, no. 21, pp. 171-192, 2022.

[7] A. Aqle, D. Al-Thani and A. Jaoua, "Conceptual interactive search engine interface for visually impaired Web users," *In 2018 IEEE/ACS 15th International Conference on Computer Systems and Applications (AICCSA) ,* vol. 15, pp. 1-6, 2018.

[8] N. A. Al-Mouh, H. S. Al-Khalifa and A. S. Al-Salman, "Proxy service to contextualize web browsing for the visually impaired," in *Proceedings of International Conference on Information Integration and Web-based Applications & Services*, 2013.

[9] F. El-Ghannam and T. El-Shishtawy, "Multi-topic multi-document summarizer," *arXiv preprint arXiv,* no. 1401.0640 , 2014.

[10] Y. J. Kumar, O. S. Goh, H. Basiron, N. H. Choon and P. C. Suppiah, "A review on automatic text summarization approaches," *Journal of Computer Science,* no. 12.4, pp. 178-190, 2016.

[11] D. Suleiman and A. Awajan, "Multilayer encoder and single-layer decoder for abstractive Arabic text summarization," *Knowledge-Based Systems ,* no. 237 , p. 107791, 2022.

[12] D. Alahmadi, A. Wali and S. Alzahrani, "TAAM: Topic-aware abstractive arabic text summarisation using deep recurrent neural networks," *Journal of King Saud University-Computer and Information Sciences ,* 2022.

[13] P. K. Wilson and J. R. Jeba, "A developed framework for multi-document summarization using softmax regression and spider monkey optimization methods," *Soft Computing ,* no. 26.7 , pp. 3313-3328, 2022.

[14] R. S. Baraka and S. N. Al Breem, "Automatic arabic text summarization for large scale multiple

documents using genetic algorithm and mapreduce," *2017 Palestinian International Conference on Information and Communication Technology (PICICT),* vol. IEEE, 2017.

[15] R. M. Badry and I. F. Moawad, "A Semantic Text Summarization Model for Arabic Topic-Oriented," *International Conference on Advanced Machine Learning Technologies and Applications. Springer, Cham,* 2019.

[16] J. M. Sanchez-Gomez , M. A. Vega-Rodríguez and C. . J. Pérez, "A multi-objective memetic algorithm for query-oriented text summarization: Medicine texts as a case study," *Expert Systems with Applications ,* no. 198 , p. 116769, 2022.

[17] N. Alami, N. En-nahnahi, S. A. Ouatik and M. Meknassi, "Using unsupervised deep learning for automatic summarization of Arabic documents," *Arabian Journal for Science and Engineering,* no. 43.12, pp. 7803-7815, 2018.

[18] R. Arumugam and R. Shanmugamani, Hands-On Natural Language Processing with Python, Birmingham: Packt Publishing Ltd, 2018.

[19] D. Jones, "Speech Recognition With Python," [Online]. Available: https://realpython.com/courses/speech-recognition-python/. [Accessed 27 November 2023].

[20] G. V. a. F. L. D. J. Rossum, The python language reference, Wilmington, DE, USA: Python Software Foundation, 2014.

[21] S. Ghosh and D. Gunning, Natural Language Processing Fundamentals, Birmingham: Packt Publishing Ltd, 2019, pp. 2-17.

[22] S. E. L. a. E. K. Bird, Natural Language Processing with Python, O'Reilly Media Inc, 2009.

[23] K. R. E. a. J. C. Taghva, "Arabic stemming without a root dictionary," *International Conference on Information Technology: Coding and Computing (ITCC'05)-Volume II,* vol. |, pp. 152-157, 2005.

[24] A. Abdi, N. Idris, R. M. Alguliyev and R. M. Aliguliyev, "Query-based multi-documents summarization using linguistic knowledge and content word expansion," *Soft Computing,* no. 21.7, pp. 1785-1801, 2017.

[25] A. Abdi, S. M. Shamsuddin and R. M. Aliguliyev, "QMOS: Query-based multi-documents opinion-oriented summarization," *Information Processing & Management ,* no. 54.2 , pp. 318-338, 2018.

[26] OpenAI, "Text to speech," 6 November 2023. [Online]. Available: https://platform.openai.com/docs/guides/text-to-speech. [Accessed 15 February 2024].

[27] "How Google's featured snippets work," [Online]. Available: https://support.google.com/websearch/answer/9351707?hl=en. [Accessed 21 October 2023].

[28] J. Achiam and e. al., "Gpt-4 technical report," arXiv preprint arxiv, 2023.

[29] R. Elbarougy, G. Behery and A. El Khatib, "Extractive Arabic Text Summarization Using Modified PageRank Algorithm," *Egyptian Informatics Journal,* 2019.

[30] Z. Li, Z. Peng, S. Tang, C. Zhang and H. Ma, "Text Summarization Method Based on Double Attention Pointer Network," *IEEE Access,* no. 8, pp. 11279-11288, 2020.

[31] M. . R. Raj, R. P. Haroon and N. SOBHANA, "A novel extractive text summarization system with self-organizing map clustering and entity recognition," *Sādhanā ,* no. 45.1 , p. 32, 2020.

[32] I. Tanfouri, G. Tlik and F. Jarray, "An automatic arabic text summarization system based on genetic algorithms," *Procedia Computer Science ,* no. 189, pp. 195-202, 2021.

[33] E. L. Pontes, S. Huet, J.-M. Torres-Moreno and A. C. Linhares, "Compressive approaches for cross-language multi-document summarization," *Data & Knowledge Engineering,* 2019.

[34] E. L. Pontes, S. Huet, J.-M. Torres-Moreno and A. C. Linhares, "Compressive approaches for cross-language multi-document summarization," *Data & Knowledge Engineering,* no. 125, p. 101763, 2020.

[35] Z. Khaleghi, M. Fakhredanesh and M. Hourali, "MSCSO: Extractive Multi-document Summarization Based on a New Criterion of Sentences Overlapping," *Iranian Journal of Science and Technology, Transactions of Electrical Engineering ,* pp. 1-11, 2020.

[36] J. M. Sanchez-Gomez, M. A. Vega-Rodríguez and C. J. Pérez., "Experimental analysis of multiple criteria for extractive multi-document text summarization," *Expert Systems with Applications ,* no. 140, p. 112904, 2020.

[37] S. Lamsiyah, A. El Mahdaouy, B. Espinasse and S. E. A. Ouatik, "An unsupervised method for extractive multi-document summarization based on centroid approach and sentence embeddings," *Expert Systems with Applications,* no. 167 , p. 114152, 2021.

[38] T. Sakai, "A Comparative Study of Deep Learning Approaches for Query-Focused Extractive Multi-Document Summarization," *2019 IEEE 2nd International Conference on Information and Computer Technologies (ICICT). IEEE,* 2019.

[39] A. Ahmad and T. Ahmad, "A Game Theory Approach for Multi-document Summarization," *Arabian Journal for Science and Engineering ,* no. 44.4, pp. 3655-3667, 2019.

[40] Y. Xu and M. Lapata, "Query focused multi-document summarization with distant supervision," *arXiv preprint arXiv:2004.03027,* 2020.

[41] D. Su, Y. Xu, T. Yu, F. B. Siddique, E. J. Barezi and P. Fung, "CAiRE-COVID: a question answering and query-focused multi-document summarization system for covid-19 scholarly information management.," *arXiv preprint arXiv:2005.03975,* 2020.

[42] P. Qi, Y. Zhang, Y. Zhang, J. Bolton and C. D. Manning, "Stanza: A python natural language processing toolkit for many human languages," *arXiv preprint arXiv:2003.07082,* 2020.

[43] C. D. Manning, An introduction to information retrieval, Cambridge university press, 2009.