International Journal of Electrical and Electronics Engineers
Volume 18, Issue No. 01, Jan-June 2026
ISSN (0) 2321-2055 ISSN (P) 2321-2045

Voice Assisted Text Summarizer Using Natural Language Processing

G. Mrudula Naga Sai Sri, D. Bhavananjali, CH. Venkatesh, CH. Rakesh Gopal,

Sk. Sadiqua Khan, Jagadeesh Thati
Department of Electronics and Communication Engineering Tirumala Engineering College, Andhra Pradesh, India

***Abstract*—In today's digital era, an enormous amount of textual information is generated daily, making it difficult for users to read and extract meaningful insights efficiently. This paper presents a Voice Assisted Text Summarizer using Natural Language Processing (NLP) techniques. The system allows users to input data in multiple formats including text, PDF, URL, and voice input. Speech input is converted into text using speech recognition techniques, followed by text preprocessing such as tokenization, stop-word removal, and normalization. The system implements extractive summarization using sentence scoring methods based on word frequency. Additionally, the summarized output is converted into speech using text-to-speech technology to enhance accessibility. The proposed system improves efficiency, reduces reading time, and supports users with accessibility needs.**

***Index Terms*—Natural Language Processing, Text Summarization, Speech Recognition, Text-to-Speech, Voice Assistant, Machine Learning**

I. Introduction

In the modern digital era, an enormous volume of textual data is generated every day through various sources such as research articles, news reports, blogs, emails, and online documents. This rapid growth of information has made it increasingly difficult for users to read, analyze, and extract meaningful insights within a limited time. As a result, there is a growing need for intelligent systems that can automatically process large amounts of text and provide concise summaries without losing essential information.

Text summarization is a key application of Natural Language Processing (NLP) that aims to reduce lengthy documents into shorter versions while preserving their core meaning. It helps users quickly understand the main ideas of a document without reading the entire content. Traditional summarization methods are often manual, time-consuming, and inefficient, especially when dealing with large datasets. Automated summarization techniques address this challenge by using computational approaches to identify and extract important information.

Despite the advancements in text summarization, most existing systems are limited to text-based input and lack accessibility features. In real-world scenarios, users may prefer voice-based interaction for convenience, especially when multitasking or for accessibility purposes. Additionally, many systems do not support multiple input formats such as PDF files or web URLs, which restricts their usability.

To overcome these limitations, this paper proposes a Voice Assisted Text Summarizer that integrates speech recognition, NLP techniques, and text-to-speech technology. The system allows users to provide input through voice, text, PDF, or URL, making it highly flexible and user-friendly. Voice input is converted into text using speech recognition, and the processed text is summarized using an extractive summarization approach based on sentence scoring.

The proposed system not only generates concise summaries but also converts them into audio output, enabling users to listen to the summarized content. This feature is particularly useful for visually impaired users and enhances overall accessibility. Furthermore, the system is designed to be lightweight and efficient, ensuring real-time performance without requiring high computational resources.

The major contributions of this work include the integration of voice-based interaction with text summarization, support for multiple input formats, and the development of a complete end-to-end system that combines NLP and speech technologies. The proposed approach improves productivity, reduces reading time, and provides a practical solution for handling information overload in the digital age.

The major contributions of this work include:

- Integration of voice input with text summarization
- Multi-format input support (text, PDF, URL)
- Efficient extractive summarization technique
- Audio output using text-to-speech

II. Literature Survey

Text summarization has evolved significantly over the years, beginning with early statistical approaches. One of the earliest contributions was made by Luhn, who introduced the concept of using word frequency to identify important sentences for summarization [1]. Later, Salton et al. extended this idea by incorporating information retrieval techniques to structure and summarize documents effectively [2]. Kupiec et al. proposed a trainable summarization model using machine learning techniques, marking a shift from purely statistical methods to data-driven approaches [3].

Further advancements introduced more sophisticated extractive techniques. Radev et al. developed centroid-based summarization for multi-document inputs, which identifies central themes across documents [4]. Similarly, Mihalcea and Tarau proposed the TextRank algorithm, a graph-based ranking model inspired by PageRank, which significantly improved sentence selection accuracy [5]. Comprehensive surveys by Nenkova and McKeown highlighted the effectiveness of extractive methods and their limitations, particularly in maintaining coherence and context [6]. Sparck Jones also provided

an overview of the evolution and challenges in automatic summarization systems [19].

With the advancement of deep learning, abstractive summarization techniques gained prominence. Nallapati et al. introduced sequence-to-sequence models using recurrent neural networks (RNNs) for generating summaries [7]. Paulus et al. further improved these models by incorporating reinforcement learning to enhance summary quality [8]. The introduction of transformer architectures by Vaswani et al. revolutionized natural language processing by enabling better contextual understanding through attention mechanisms [9]. Devlin et al. proposed BERT, which significantly improved language representation and downstream tasks including summarization [10]. Liu and Lapata demonstrated the effectiveness of pretrained transformer models for summarization tasks [11].

In addition to summarization, speech processing plays a crucial role in voice-assisted systems. Traditional speech recognition techniques were introduced by Rabiner and Juang [15], while Graves et al. later applied deep learning methods such as recurrent neural networks to improve speech recognition accuracy [16]. Modern APIs such as Google Text-to-Speech and AssemblyAI provide efficient solutions for speech synthesis and recognition, enabling real-time voice interaction in applications [17], [18].

Supporting technologies such as word embeddings and natural language processing libraries have also contributed to system development. Mikolov et al. introduced word embedding techniques that capture semantic relationships between words [12]. Bird et al. provided practical tools for NLP implementation through the NLTK library [13]. Foundational concepts in information retrieval and NLP are well-documented by Manning et al. and Jurafsky and Martin, which guide the development of modern summarization systems [20], [14].

Despite these advancements, existing systems often lack integration of voice-based interaction with summarization capabilities. The proposed system addresses this gap by combining NLP-based summarization with speech recognition and text-to-speech technologies to provide a more accessible and efficient solution. .

III. Problem Statement

Existing systems lack:

- Voice interaction
- Multi-format input
- Real-time processing

A system is needed that provides efficient summarization with accessibility.

IV. Proposed System

The proposed system is a Voice Assisted Text Summarizer designed to efficiently process large volumes of textual data and generate concise summaries. Unlike traditional summarization systems, the proposed model supports multiple input formats including plain text, PDF documents, web URLs, and voice input. This flexibility allows users to interact with the system in a convenient and user-friendly manner, making it suitable for real-world applications such as education, research, and content analysis.

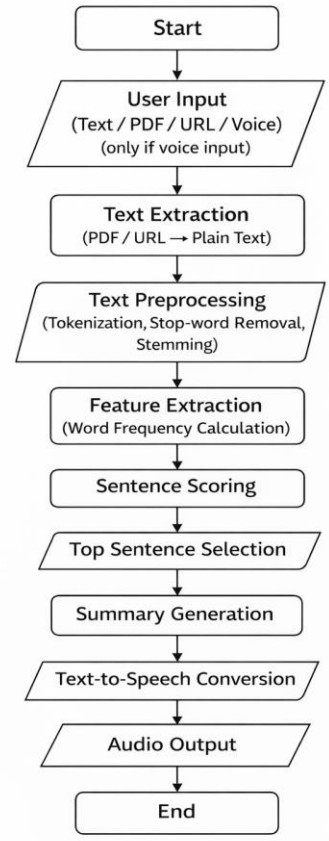

Fig. 1. Proposed System

The system integrates speech recognition, Natural Language Processing (NLP), and text-to-speech technologies to provide a complete end-to-end solution. When the user provides voice input, it is first converted into text using a speech recognition module. The extracted or input text is then preprocessed using NLP techniques such as tokenization, stop-word removal, and normalization. These steps ensure that the data is clean and suitable for further processing.

An extractive summarization approach is used to generate summaries by identifying and selecting the most relevant sentences from the input text. Sentence importance is determined using a frequency-based scoring mechanism, where sentences containing high-frequency keywords are given higher priority. This approach ensures that the generated summary retains the essential information while reducing redundancy.

Finally, the summarized text is converted into audio output using text-to-speech technology, enhancing accessibility for users, especially those with visual impairments. The system is implemented using lightweight tools and frameworks, ensuring fast processing and real-time performance. Overall, the

proposed system provides an efficient, accessible, and scalable solution for automatic text summarization.

A. Speech Recognition

Voice input is converted into text using speech recognition APIs such as Assembly AI.

B. Text Preprocessing

The input text is processed using:

- Tokenization
- Stop-word removal
- Stemming
- Lowercasing

C. Feature Extraction

Important features such as word frequency are calculated to determine sentence importance.

D. Sentence Scoring

Each sentence is assigned a score based on frequency of important words.

E. Summary Generation

Top-ranked sentences are selected to generate the summary

V. Methodology

A. Speech Recognition

Voice input provided by the user is converted into textual data using a speech recognition module. This module processes the audio signal, removes background noise, and identifies spoken words using trained models. The converted text serves as the primary input for further natural language processing tasks.

B. Preprocessing

The extracted text undergoes preprocessing to improve its quality and consistency. Tokenization is performed to split the text into individual words or sentences. Stop-word removal eliminates commonly used words that do not contribute significant meaning. Stemming is applied to reduce words to their root form, and all text is converted to lowercase to maintain uniformity.

C. Feature Extraction

In this stage, important features are identified from the processed text. Word frequency is calculated to determine how often each word appears in the document. Frequently occurring words are considered more important and are used as key indicators for identifying relevant sentences.

D. Sentence Scoring

Each sentence in the document is assigned a score based on the frequency of important words it contains. Sentences with higher occurrences of significant keywords receive higher scores. This scoring mechanism helps in identifying sentences that best represent the overall content of the document.

E. Summary Generation

The final summary is generated by selecting the top-ranked sentences based on their scores. These sentences are arranged in their original order to maintain coherence and readability. The resulting summary provides a concise representation of the original text while preserving its key information.

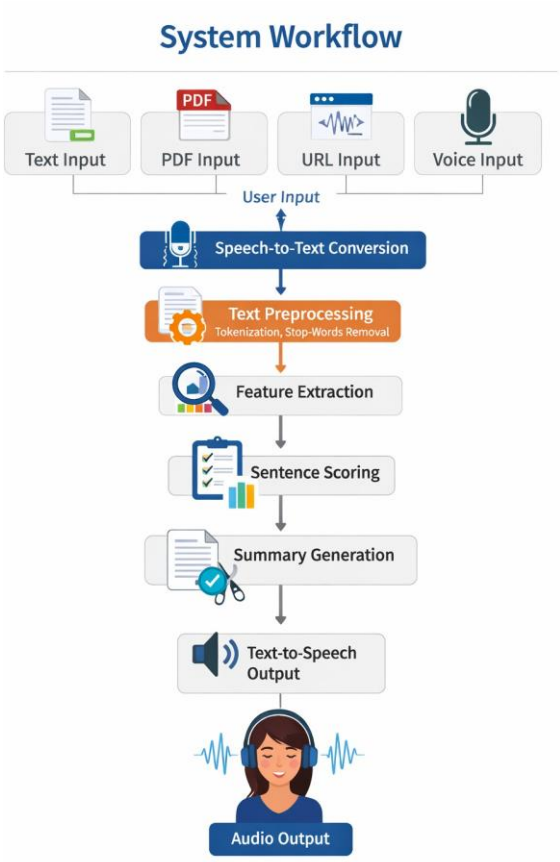

Fig. 2. System Workflow

VI. Tools and Technologies

- Python
- NLTK
- PyPDF2
- BeautifulSoup
- Assembly AI
- Google Text-to-Speech
- Streamlit

VII. Implementation

The system is implemented in Python. Text is extracted from PDFs and URLs. Voice input is converted to text. NLP techniques are applied to preprocess data. The summary is generated and converted to speech using gTTS. Streamlit provides the interface.

VIII. Results and Discussion

The proposed Voice Assisted Text Summarizer was evaluated using different types of input sources such as plain text, PDF documents, web URLs, and voice input. The system successfully processed all input formats and generated concise summaries while preserving key information. The integration of speech recognition and text-to-speech modules enabled seamless interaction, making the system user-friendly and accessible.

A. Performance Metrics

To evaluate the effectiveness of the summarization process, the following metrics were used:

$$\text{Compression Ratio} = \frac{\text{Length of Summary}}{\text{Length of Original Text}} \quad (1)$$

$$\text{Precision} = \frac{\text{Relevant Sentences}}{\text{Selected Sentences}} \quad (2)$$

$$\text{Recall} = \frac{\text{Relevant Sentences}}{\text{Total Relevant Sentences}} \quad (3)$$

$$F1\ \text{Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (4)$$

B. Experimental Results

The system was tested on multiple datasets including articles, research papers, and web content. Table I shows the performance evaluation of the proposed system.

TABLE I
Performance Evaluation of Proposed System

Input Type	Comp. Ratio	Precision	Recall	F1 Score
Text Document	0.35	0.82	0.78	0.80
PDF Document	0.38	0.80	0.76	0.78
Web URL	0.33	0.84	0.79	0.81
Voice Input	0.40	0.78	0.74	0.76

C. Discussion

From the results, it is observed that the system achieves a good balance between compression and information retention. The compression ratio ranges between 0.33 and 0.40, indicating that the system effectively reduces the text size while maintaining important content. The precision and recall values show that most of the selected sentences are relevant and significant.

The performance for text and URL inputs is slightly higher compared to voice input due to minor inaccuracies in speech recognition. However, the system still provides acceptable results for voice-based summarization, demonstrating its robustness.

The use of extractive summarization ensures faster processing time and lower computational requirements compared to deep learning-based abstractive methods. However, it may sometimes lack coherence and contextual flow in the generated summary.

D. Graphical Analysis

The performance comparison of different input types is illustrated in Fig. 3. It shows that text and URL inputs achieve higher precision and F1 scores, while voice input slightly lags due to speech-to-text conversion errors.

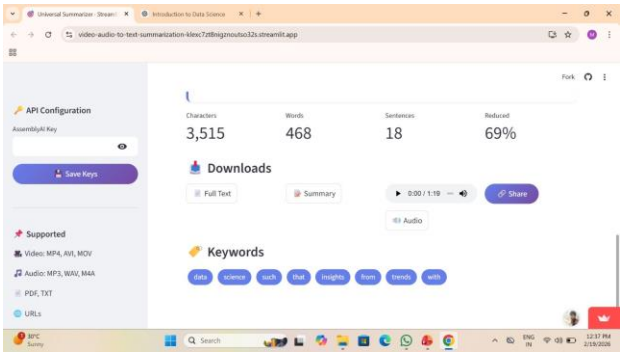

Fig. 3. Output

Overall, the proposed system demonstrates reliable performance across multiple input formats. It effectively integrates NLP and speech technologies, providing a practical solution for real-world applications such as education, content analysis, and accessibility support.

IX. Advantages

- Saves time
- Reduces reading time
- Supports voice interaction
- User-friendly interface
- Multi-format support
- Improves productivity for students, researchers, and professionals
- Enables quick understanding of large documents
- Helpful for visually impaired users through audio output
- Eliminates the need to manually read long documents
- Real-time processing provides instant summaries
- Lightweight system compared to deep learning models
- Does not require high computational resources
- Easy to deploy as a web or mobile application

X. Limitations

- May miss context
- English only

XI. Conclusion

The system improves productivity and accessibility by combining NLP and speech technologies.

XII. Future Work

- Deep learning integration
- Multi-language support
- Mobile app

REFERENCES

[1] H. P. Luhn, "The automatic creation of literature abstracts," IBM Journal of Research and Development, vol. 2, no. 2, pp. 159–165, 1958. doi: 10.1147/rd.22.0159
[2] G. Salton, A. Singhal, M. Mitra, and C. Buckley, "Automatic text structuring and summarization," Information Processing & Management, vol. 33, no. 2, pp. 193–207, 1997. doi: 10.1016/S0306-4573(96)00062-3
[3] J. Kupiec, J. Pedersen, and F. Chen, "A trainable document summarizer," in Proc. SIGIR, 1995, pp. 68–73. doi: 10.1145/215206.215333
[4] D. R. Radev, H. Jing, M. Stys´, and D. Tam, "Centroid-based summarization of multiple documents," Information Processing & Management, vol. 40, no. 6, pp. 919–938, 2004. doi: 10.1016/j.ipm.2003.10.006
[5] R. Mihalcea and P. Tarau, "TextRank: Bringing order into texts," in Proc. EMNLP, 2004, pp. 404–411.
[6] S. Nenkova and K. McKeown, "Automatic summarization," Foundations and Trends in Information Retrieval, vol. 5, no. 2–3, pp. 103–233, 2011. doi: 10.1561/1500000015
[7] A. Nallapati, B. Zhou, C. Gulcehre, and B. Xiang, "Abstractive text summarization using sequence-to-sequence RNNs," arXiv preprint arXiv:1602.06023, 2016.
[8] R. Paulus, C. Xiong, and R. Socher, "A deep reinforced model for abstractive summarization," arXiv preprint arXiv:1705.04304, 2017.
[9] A. Vaswani et al., "Attention is all you need," in Proc. NIPS, 2017, pp. 5998–6008. doi: 10.48550/arXiv.1706.03762
[10] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers," arXiv preprint arXiv:1810.04805, 2018.
[11] Y. Liu and M. Lapata, "Text summarization with pretrained encoders," in Proc. EMNLP, 2019. doi: 10.18653/v1/D19-1387
[12] T. Mikolov et al., "Efficient estimation of word representations in vector space," arXiv preprint arXiv:1301.3781, 2013.
[13] S. Bird, E. Klein, and E. Loper, "Natural Language Processing with Python," O'Reilly Media, 2009.
[14] D. Jurafsky and J. H. Martin, "Speech and Language Processing," 3rd ed., Pearson, 2023.
[15] L. Rabiner and B. Juang, "Fundamentals of speech recognition," Prentice Hall, 1993.
[16] A. Graves et al., "Speech recognition with deep recurrent neural networks," in Proc. ICASSP, 2013. doi: 10.1109/ICASSP.2013.6638947
[17] Google, "Google Text-to-Speech API Documentation," 2024. [Online]. Available: https://cloud.google.com/text-to-speech
[18] AssemblyAI, "Speech-to-Text API Documentation," 2024. [Online]. Available: https://www.assemblyai.com
[19] K. Sparck Jones, "Automatic summarising: The state of the art," Information Processing & Management, vol. 43, no. 6, pp. 1449–1481, 2007. doi: 10.1016/j.ipm.2007.03.009
[20] C. D. Manning, P. Raghavan, and H. Schu¨tze, "Introduction to Information Retrieval," Cambridge University Press, 2008.