ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26aug453

AI-Driven Kannada Document Summarization Using Optical Character Recognition and Natural Language Processing: A Web-Based Implementation Framework

Apoorva S.[1]; Usha B. S.[2]; S. M. Darshan[3]

[1]Department of Computer Science and Engineering, East West Institute of Technology, Bengaluru, Karnataka, India
[2]Department of Computer Science and Engineering, East West Institute of Technology, Bengaluru, Karnataka, India
[3]Department of Mechanical Engineering, School of Engineering and Technology, CHRIST (Deemed to be University), Bengaluru, Karnataka, India

Publication Date: 2026/08/21

Abstract: The fast expansion of digital data has resulted in high demand for smart systems which can quickly obtain compact and meaningful data from lengthy documents. Even though automation in text summarization has seen significant developments in well-resourced languages, automated summarization of Kannada is still rare due to the intricacies of the script of the language, poor computational power, and massive information available in printed and scanned form. A web-based framework for automated summarization of Kannada documents is introduced in this paper, utilizing AI by developing a processing platform that integrates OCR and NLP. This technique works not only with originally typed text in Kannada but also with scanned documents. Scanned documents are converted to editable Unicode by using Tesseract OCR engine before performing the language-specific NLP tasks that include normalization, tokenization, sentence splitting, stopword removal, and extraction of summaries. Using technologies such as Python, Flask, OpenCV, Tesseract OCR, and relational database management, the developed application can ensure secure authentication of users, management of documents, and visualization of summaries using an interactive web interface. The successful tests indicate that OCR and NLP technologies have been integrated into the process of performing various tasks related to the examination of documents written in the Kannada language. The modular architecture of the project enables applying transformer-based summaries, document processing in many languages, OCR of handwritten texts written in the Kannada language, and various technologies for running applications in the cloud in the future. Thus, the developed application is an example of the effective use of Artificial Intelligence in processing documents in regional languages and lays the groundwork for creating automated systems for document management.

Keywords: *Artificial Intelligence, Optical Character Recognition, Natural Language Processing, Kannada Document Summarization, Extractive Summarization, Tesseract OCR, Flask, Intelligent Document Processing.*

How to Cite: Apoorva S.; Usha B. S.; S. M. Darshan (2026) AI-Driven Kannada Document Summarization Using Optical Character Recognition and Natural Language Processing: A Web-Based Implementation Framework. *International Journal of Innovative Science and Research Technology*, 11(8), 1225-1235. https://doi.org/10.38124/ijisrt/26aug453

I. INTRODUCTION

➢ *Background*

Information and Communication Technology (ICT) has acquired a significant development, which has changed the way textual information is being created, stored, and shared. The establishments of education, governmental organizations, healthcare systems, research facilities, law departments, and even media establishments continuously produce a huge bulk of documents in electronic formats like report, books, journals, official documentation, newspapers, manuals, and law materials. Nonetheless, despite the fact that the digital transformation has helped the society considerably in terms of access to information, the huge amount of produced text poses questions of retrieval of necessary information easily, since the amount of data grows each day [1]–[3]. Automatic text summarization is one of the useful aspects of NLP that allows long text documents to be transformed into short ones while retaining key contextual information. Summarization techniques are typically divided

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26aug453

into two main categories: extractive and abstractive. Extractive summarization specifies the most effective sentences from the source document through the use of different statistical, linguistic, or semantic characteristics. Abstractive summarization, on the other hand, is responsible for forming new sentences to express the meanings of the original text. Although performance in terms of abstractive summarization has been greatly enhanced by transformer architectures for resource-rich languages, such methods are not quite functional for low-resource languages like Kannada since they lack sufficient datasets, computing resources for language processing, and pre-trained language models. As a result, extractive summarization is still the most efficient and applicable method for dealing with documents in regional languages [4]–[6].

The Kannada language is one of the major Dravidian languages and is the primary language of millions of people in areas such as education, administration, judiciary, healthcare, media, and literature. Despite digitization initiatives, a large amount of important Kannada information is still available in print formats like books, newspapers, historical texts, government documents, magazines, and images. In order to retrieve information from these media effectively, intelligent computational processes are required to deal with both machine-typed and image-based text. Unfortunately, most readily available summarization systems work well only for the English language and a handful of important languages, creating a gap in the technology available for Kannada document processing. The solution lies in exploring the ways in which language-specific AI technologies are able to combine document digitization and summarization processes to make regional document information more accessible [7]–[9].

OCR or Optical Character Recognition provides a useful method for converting Kannada documents from paper to editable text. But the accurate understanding of Kannada characters still seems to be a hard task to accomplish because of the intricate nature of Kannada script, wrestling with conjunct consonants and vowel modifiers, many different styles of fonts, etc. In order to achieve better results ordinary image preprocessing operations like grayscale conversion, adaptive thresholding, skew correction and noise reduction are implemented first for improving accurateness. The text is then processed through NLP systems [10]–[12].

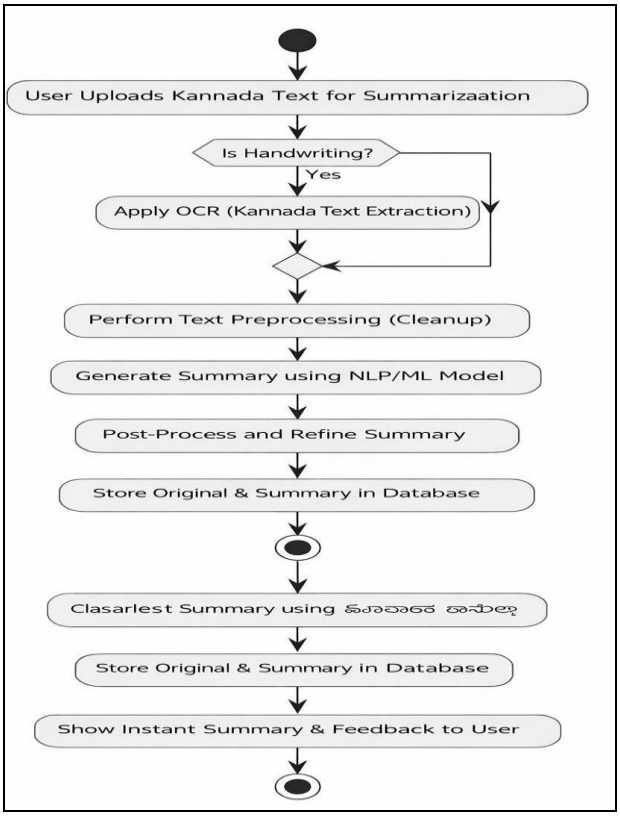

Fig 1 Overall Architecture of the Proposed AI-Driven Kannada Document Summarization Framework.

The nuts and bolts of the AI-based document summarization system are depicted in Fig. 1. The system is a combination of document acquisition, OCR, NLP, database management, and a web user interface in a modular structure which enables it to handle both typed and scanned Kannada documents. The modularity also allows each processing part of the system to work independently and paves the way for future developments like multilingual documents processing, transformer-based summarization, OCR of handwritten Kannada documents, and cloud implementation.

The possible uses of the framework are outlined in Table 1. The system is meant for helping schools, government agencies, medical institutions, legal entities, libraries, research institutions, and media organizations by providing a means to get required data from large volumes of Kannada texts quickly and easily.

Table 1 Applications of AI-Based Kannada Document Summarization

Application Domain	Typical Documents	Expected Benefit
Education	Notes, textbooks, assignments	Faster learning
Government	Circulars, notifications	Quick decision making
Healthcare	Medical reports	Faster information retrieval
Legal	Judgements, legal documents	Reduced review time
Media	Newspapers, articles	Rapid news summarization
Research	Research papers	Efficient literature review

➢ *Research Motivation*

The increasing use of digital governance, online learning, and electronic document management has resulted in the enhanced volumes of the Kannada textual content generated in both digital and scanned formats. However, effective retrieval of useful knowledge from the documents

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26aug453

in a regional language continues to remain a big issue due to the lack of computational capabilities, errors in OCR, and absence of NLP tools for the languages. Most existing document processing tools consider OCR and text content extraction as independent tasks, which does not provide the needed practical usability of solutions in terms of working with the documents that existed as scanned and typed before. As a result, developing a unified framework that combines OCR and NLP within one web application should help make the documents accessible, facilitate the reduction of labor input, and make it possible to implement AI in Kannada [13]-[15].

➢ *Problem Statement*

Despite many advances in Optical Character Recognition and automatic text summarization, implementations generally do not allow fully functional systems where users can upload both typed Kannada text and images or scanned documents through a user-friendly web interface. Moreover, existing systems often lack much-needed document management capabilities, modular design, and easy implementation that are necessary to install practical AI applications. This necessitates the development of an integrated AI platform that unites Optical Character Recognition, Natural Language Processing, and web technologies for the needs of efficient and scalable document summarization for documents in Kannada [16]–[18].

➢ *Major Contributions*

The key achievements of the present work are summarized as follows:

- Creation of a cohesive AI-centered system for summarizing Kannada textual documents.
- Utilizing Optical Character Recognition (OCR) technology in processing Kannada personal images and PDFs.
- Usage of language-focused Natural Language Processing (NLP) and extractive techniques aimed at summarizing text contents.
- Implementation of a safe web application by utilizing Python, Flask, OpenCV, and Tesseract OCR.
- Verification of the functionality of the developed system based on typed and scanned text papers.
- Modular design which can be enhanced in the future by adding features such as transforming models and cloud computing.

II. RELATED WORK

In recent years, technological advancements in Artificial Intelligence (AI), Optical Character Recognition (OCR), and Natural Language Processing (NLP) have boosted the functionality of intelligent document processing systems by facilitating text extraction, semantic analysis, and automatic summarization. OCR technologies have come a long way from traditional methods of pattern recognition to the adoption of deep learning technologies that enable recognition of documents in a variety of languages with more precision. Likewise, NLP technologies have evolved from the use of statistical models of language to the new transformer architecture used in NLP today, which has led to the improved performance of document understanding and summarization tools. Research on Kannada document summarization has fallen behind, though it is critical that an integrated OCR-NLP platform be designed that suits the needs of Kannada language users since Kannada language is highly morphologically complex, making it difficult to find annotated datasets needed for development of effective OCR and NLP technologies for this language [19]–[22].

➢ *Optical Character Recognition for Kannada Documents*

OCR is a crucial technology used in digitizing printed documents or scanned images. Modern OCR systems make use of various techniques like image preprocessing, feature extraction, and deep learning to enhance its accuracy in recognizing different document types. Tesseract OCR is one of the most popular engines because of its multilingual features, open source nature, and compatibility with libraries like OpenCV. Despite being an easy process with other languages, OCR for Kannada is quite difficult due to the nature of language which has agglutination, character compounds, conjunct consonants, different vowel modifiers, and various font types. Recent research shows that many preprocessing methods such as gray scaling, adaptive thresholding, skewing, and noise cancellation have a great effect on the accuracy of OCR [23]–[27].

➢ *Natural Language Processing and Automatic Text Summarization*

NLP (Natural Language Processing) is essential for gaining valuable insights from textual documents via language comprehension, syntactic parsing, semantic analysis, and automatic summarization. Summarization strategies can be broadly classified into extractive and abstractive categories. The extractive technique selects important sentences from the original document through statistical and linguistic characteristics of the text while the abstractive technique uses neural networks to produce meaningful summaries. While models such as BERT, T5, PEGASUS, and BART have improved the efficacy of the abstractive summarization technique, they have not been successfully implemented in the case of Kannada because of language constraints and the level of computing power needed. Therefore, the extractive approach remains the only practicable and cost-effective technique for processing Kannada texts. In addition, researchers are now trying to incorporate OCR (optical character recognition)-generated text into the NLP system for intelligent text extraction from scanned regional language documents [28]–[32].

➢ *Research Gap*

While there has been a considerable advancement in OCR, NLP, and automatic summary creation, not many studies have been done to use these technologies effectively for the purpose of Kannada document summarization. Most of the existing work has either concentrated on utilizing OCR for document conversion or on turning already converted text into summaries disregarding high potential of developing the system which can work with both written and typed Kannada documents within one web-based program. Also, some

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26aug453

available solutions lack such important features as secure document management, modular software architecture, and implementation of proper measures to run it. The research conducted proposes a new AI-based framework for performing OCR, NLP, and summary creation. Table 2 presents the comparison of some studies with respect to the gaps that were identified during the research process [33]–[34].

Table 2 Comparative Analysis of Existing Studies and Identified Research Gaps

Existing Limitation	Impact	Proposed Solution
Digital text required	Cannot process scanned documents	OCR-based text extraction
Limited Kannada NLP resources	Reduced summarization quality	Language-specific preprocessing
Standalone OCR applications	No automated summarization	Integrated OCR and NLP
Research-oriented prototypes	Limited practical usability	User-friendly web application
High computational requirements	Difficult deployment	Lightweight modular implementation

The shortcomings found in the literature analyzed led to the elaboration of the framework presented in the next chapter. In contrast to traditional technologies that carry out OCR or summarization separately, the proposed model provides functionality for document acquisition, OCR, NLP, extractive summarization, database administration, and a web interface based on Flask in one implemented platform developed for the processing of documents in the Kannada language.

III. PROPOSED FRAMEWORK AND METHODOLOGY

As seen in the previous sections of the study, existing literature has shown that OCR, NLP, and automatic summarization have been primarily explored as individual processes. It is, however, worth mentioning that very few studies have come up with an integrated solution for the processing of scanned and typewritten Kannada text. Therefore, in light of the problems highlighted, the current study intends to propose a web application based on AI that encompasses OCR, NLP, extractive summarization, database management, and user interaction capabilities. The web application is aimed at automating the entire document processing pipeline, thus minimizing the efforts undertaken and making access to information in Kannada more straightforward [18], [24], [31].

➢ *Overall System Architecture*

The proposed framework employs a modular design consisting of six different functional components, namely, document acquisition, image preprocessing, OCR, NLP, summary generation, and database management that completes a specific job while sending data between them by means of well-defined interfaces. This increases modularity, maintainability, and scalability of the framework. The framework accepts both typed Kannada text as well as scanned image/PDF documents and thus allows using different documents at the same platform. This modular approach allows the framework to easily incorporate advanced AI technologies in the future [12], [18], [24].

Table 3 Functional Modules of the Proposed Framework

Layer	Major Components	Primary Function
Presentation Layer	Flask, HTML, CSS, Bootstrap	User interaction
Input Layer	Text Entry, Image Upload, PDF Upload	Data acquisition
OCR Layer	Tesseract OCR, OpenCV	Text extraction
NLP Layer	Tokenization, Stop-word Removal, Sentence Ranking	Summary generation
Database Layer	SQLite	Data storage
Output Layer	Summary Display	Result presentation

The first step the user has to do is authenticate his/her identity using a web application and upload the necessary documents for processing. Typed Kannada texts move forward to the NLP module, while scanned documents are first preprocessed and their content extracted by means of OCR. The extracted Unicode text then goes through a language-specific preprocessing phase before the summarization produces a final output. The produced results are stored in a database and used in future by the means of a web interface [22], [27], [31].

The complete design of the framework has the interconnection and interaction of various components, which include the document acquisition module, the OCR engine, the Natural Language Processing module, the database, and the presentation layer. The methodology allows the software modules to operate independently while providing a successful, progressive document workflow.

The responsibilities of each software module are presented in Table 3, outlining their capabilities within the overall framework.

➢ *System Workflow*

A sequential processing flow is employed in the proposed workflow in order to facilitate efficient processing of both digital and scanned documents in Kannada. The users send either a typed text or a scanned image/PDF through GUI after completing a successful login. The digitized text does not go through OCR and is directly passed to the

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26aug453

Natural Language Processing component. Unlike the digital documents, the scanned documents undergo the image enhancement process wherein the optical character recognition extracts a usable text in Unicode Kannada from the scanned images [19], [23], [30].. The extracted text is then normalized, tokenized, segmented, and then analyzed for generating a summary. Finally, the summary is made available to the users and saved in the database for future reference.

The process is shown in Fig. 2 where the workflow is illustrated and the interaction between all modules taking place in the workflow is shown explicitly in the image.

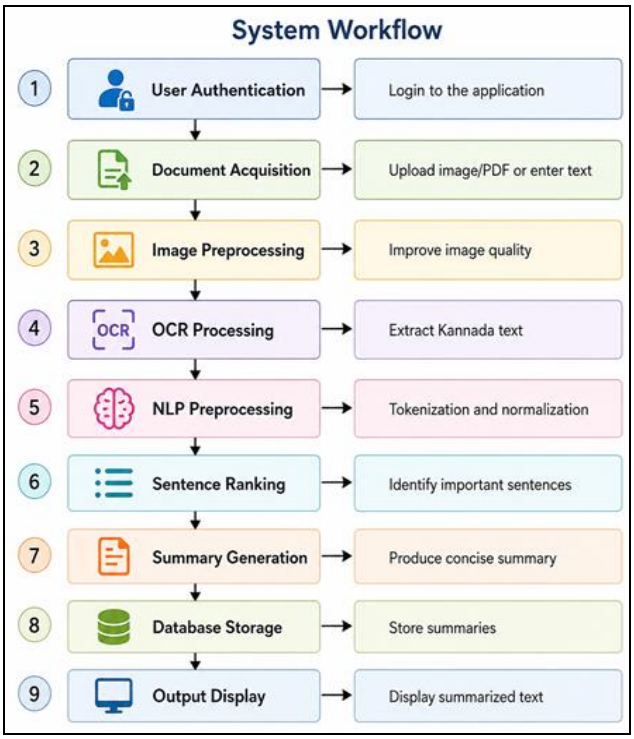

Fig 2 Workflow of the Proposed Kannada Document Summarization Framework.

➢ *OCR Processing Pipeline*

Optical Character Recognition represents the most important component facilitating the process of converting scanned Kannada documents into editable text. Due to the fact that the quality of the scanned document affects recognition accuracy, image preprocessing is performed prior to OCR application. The preprocessing stage includes gray scaling, adaptive thresholding, noise elimination, skew correction, image enhancement aimed at improving visibility of the text while decreasing recognition errors caused by inadequate illumination, scanning artifacts, and deterioration of documents. These actions significantly enhance the quality of the extracted text prior to starting its processing by NLP [16], [20], [24].

After preprocessing is done, the enhanced image undergoes Tesseract OCR engine processing resulting in Unicode Kannada text being extracted for further language processing purposes. The extracted text moves further to the NLP module automatically without any manual assistance from the operator. In this way, document digitization and summarization processes are completely automated in one framework [21], [24], [31]. The final scheme of OCR processing pipeline applied in the suggested framework is demonstrated in Fig. 3. All these elements of the framework form a reliable pipeline for document digitization suitable for Kannada documents.

Stage	Operation	Purpose
Image Upload	Receive scanned image/PDF	Document acquisition
Image Preprocessing	Grayscale, thresholding, denoising	Improve image quality
Character Recognition	Tesseract OCR	Extract Kannada text
Text Validation	Remove OCR artefacts	Improve text quality
Text Output	Unicode Kannada text	Input for NLP

Fig 3 OCR Processing Pipeline for Kannada Document Digitization.

➢ *Natural Language Processing Workflow*

In the preprocessing phase, language-specific techniques are applied to the extracted Kannada text to ensure the quality of the generated summary. The preprocessing operations consist of text normalization, tokenization, sentence segmentation, removal of stop-words, and cleaning methods which are used to reduce excessive linguistic noise as well as preserve the semantic integrity of the document.

Table 4 Natural Language Processing Components

Stage	Description
Text Normalization	Remove unwanted symbols and formatting
Sentence Segmentation	Divide text into sentences
Tokenization	Split sentences into words
Stop-word Removal	Remove insignificant words
Sentence Ranking	Determine sentence importance
Summary Generation	Produce concise summary

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26aug453

The processed text is then analyzed using the extractive summarization technique that ranks the individual sentences according to their importance and chooses the most informative sentences for the final summary. The reason for adopting this approach is its computational efficiency and ability to process Kannada language documents [22], [26], [32]. The main components of the language processing are presented in Table 4.

➢ *Proposed Methodology*

The full methodology used in the proposed framework integrates document acquisition, Optical Character Recognition (OCR), Natural Language Processing (NLP), extractive summarization, database management, and web-based visualization in a seamless processing pipeline. At first, authorized users can upload their input text document in either handwritten Kannada or scanned document form through an application interface. Then, scanned documents are text-processed through OCR while text documents are forwarded directly to the Natural Language Processing module. After processing the language and preparing the summary through extractive summarization, the summary is displayed via the web interface and stored in a relational database at the same time. This repetitive process enables the successful processing of various types of Kannada documents along with less manual work and higher accessibility of information in the original Kannada.

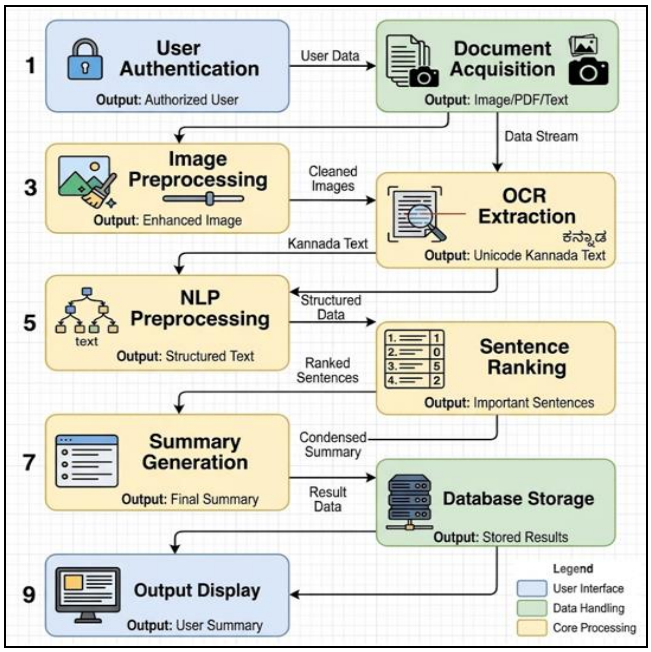

Fig 4 Proposed Methodology for AI-Driven Kannada Document Summarization.

The methodology is shown in Fig. 4, which shows how everything works together and interacts from document acquisition to summary production. One more advantage of the developed architecture is that it allows for future adaptations such as multilingual processing, transformer-based summarization, handwritten Kannada OCR, and cloud computing.

IV. SYSTEM IMPLEMENTATION

The AI-driven Kannada document summarizer has been implemented as a web application by combining Optical Character Recognition (OCR), Natural Language Processing (NLP), and database management technology into a single software package. The way of implementation follows the modular architecture explained in Section III, where each module functions independently and exchanges data among each other through strict interfaces. Such an approach allows improving software maintenance, scalability, and expandability in the future without compromising the operation of the system as a whole [18], [24], [31].

The application implements secure user authentication, document uploading, OCR-based text extraction from scanned documents, Kannada language processing, extractive summarization, and summary visualization through a standard web interface.

The application was developed in Python which is the best programming language when it comes to Artificial Intelligence, image processing, and Natural Language Processing technologies. Furthermore, Flask was chosen as the web application framework due to its lightweight architecture and ease of integration. OpenCV was used for image preprocessing, while Tesseract OCR engine was used in order to recognize Kannada text from the documents. The summarization module uses language-dependent preprocessing and extractive summarization techniques to summarize the extracted texts.

A relational database management system was used for the keeping of user credentials, the uploaded documents and the summaries produced, which allowed the effective functioning of document management system and its future retrieval [12], [20], [27].

➢ *Development Environment*

The implementation environment was selected to ensure platform independence, simplicity of setup and compatibility with the technologies of open source. Python has numerous libraries for Artificial Intelligence and Natural Language Processing, while Flask allows creating scalable web applications with few computing resources used. Image preprocessing operations were done with the help of OpenCV, while the implementation of OCR was performed using Tesseract OCR engine that was configured to recognize Kannada Unicode.

Table 5 Software and Hardware Configuration of the Developed System

Component	Technology Used
Operating System	Windows 10 / 11
Programming Language	Python 3.x

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26aug453

Web Framework	Flask
OCR Engine	Tesseract OCR
OCR Library	Pytesseract
Image Processing	OpenCV
NLP Libraries	NLTK / spaCy
Database	SQLite
IDE	Visual Studio Code
Browser	Google Chrome / Microsoft Edge

The database operations were conducted with the help of a relational database that stores all user data, uploaded documents, the outputs of OCR and the generated summaries. The software and hardware configuration used during implementation is provided in Table 5.

➢ *User Authentication Module*

The user authentication module ensures the secure operation of the application by authenticating registered users before the document processing starts. It offers the functionality of user registration, secure login, password checking, and session management that prevents unauthorized access to the uploaded documents and knowledge bases summarization output. After the successful authentication, the user is redirected to the dashboard of the application to upload and process the Kannada documents. This module provides the first layer of security in the system and controlled access to the document management services [18], [24], [30].

➢ *OCR Module Implementation*

OCR technology is utilized to manage the scanned Kannada paper documents submitted by authorized users. Following the document upload, image preprocessing processes may be conducted, such as converting images to grayscale, adaptive thresholding, skew correction, and noise reduction, for improving the quality of images utilized in Optical Character Recognition. Then the processed image will be analysed by the Tesseract OCR engine, which transforms the document into Unicode Kannada text that can be easily edited. Then, the derived text will automatically be transferred to the NLP module, thereby ensuring that no human involvement is required to digitize the document.

➢ *Kannada Document Summarization Module*

Once the Kannada text has been extracted through OCR or by means of direct text input, it goes through preprocessing which is special for that language before being summarized. The implementation normalizes, tokens the text, segments it into sentences, and removes stop-words before extracting summaries and improving its quality. After that, it finds important sentences and produces a summary without losing the main essence of the text.

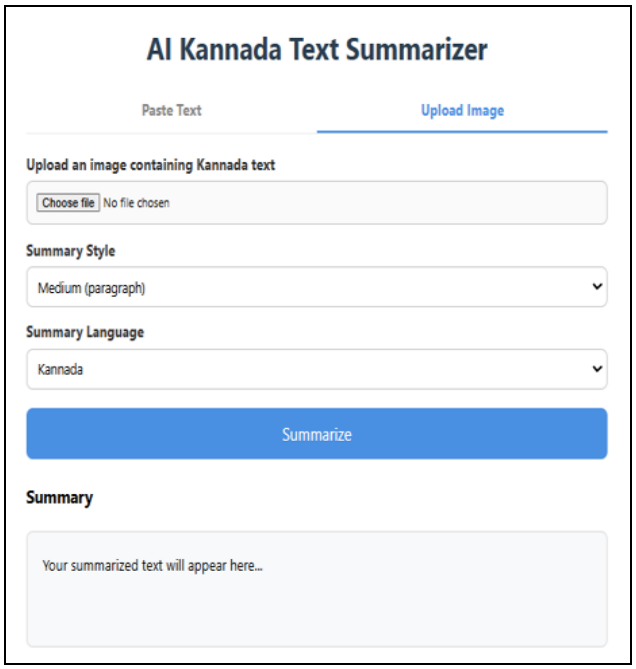

Fig 5 Kannada Document Summarization Interface.

The modular implementation makes it possible for the summarization part of the tool to process both OCR-produced and digitally produced Kannada texts through the same processing chain which increases both flexibility of the software and simplicity of its implementation. The summarization interface provided by the developed software is shown in Fig. 5.

➢ *Summary Visualization and Database Management*

The summary created along with the document information is saved in an efficient manner to ensure that it can be accessed easily. All the user details, document details, OCR results, summary, and history of processing are saved in an organized manner and may be easily retrieved when needed. The summary visualization module presents the created summary of the document using an easy-to-use mode on the web and allows the user to manage their documents easily. Also, the developed system uses a combination of the database technology with OCR and Natural Language Processing to ensure that the developed system can be easily used by educational institutions, government bodies, research labs, and digital libraries [18], [24], [31].

V. RESULTS AND DISCUSSIONS

The implemented framework is designed using Artificial Intelligence (AI) techniques and has undergone effectiveness assessment to ensure that Optical Character

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26aug453

Recognition (OCR), Natural Language Processing (NLP), and web-based document management are thoroughly integrated. Specifically, the assessments aimed to check that the flow of document processing was efficient in terms of the stages involved such as user authentication to document upload, OCR-poised text extraction to the processing of documents in Kannada language, extractive summarization, and storage of files in databases. The tests carried out showed that the developed system successfully processes both typed text in Kannada and scanned document/image files at a time. The modular architecture of the solution conveyed the feasibility of this framework in terms of its realization and allowed the individual components to work together [18], [24], [31].

➢ *Experimental Setup*

The application was examined in accordance with the execution atmosphere as explained in Section IV of this work. There were multiple functional tests in this work by using a set of Kannada documents containing either text printed electronically or images coming from scanned documents in various common formats that can be seen in educational institutions, government offices, and public information systems. The evaluation was aimed primarily at determining operational reliability of certain modules of software, how well all the processing components work together and the ability of the framework to produce meaningful summary regardless of the input document types. The goal of this work is to provide evidence of the ability to create AI system based on the integrated solution; that is why the functional performance of the application was of primary importance and comparison with existing algorithms [20],[27],[32] was not an issue. Table 6 details the datset type and its description for the functional test cases performed on the developed application.

Table 6 Test Dataset

Dataset Type	Description
Typed Kannada Text	Manually entered text
Printed Kannada Document	High-quality scanned pages
PDF Document	Multi-page Kannada PDF
Image Document	JPEG/PNG Kannada image

➢ *Functional Validation*

The process of functional validation was undertaken to confirm that every module was functioning correctly in the proposed framework. In the initial step, authorized users typed in Kannada or uploaded scanned documents through the web interface. For scanned documents, the OCR module was able to convert the photographs into editable text after image preprocessing, while the typed Kannada documents were processed directly by the NLP module.

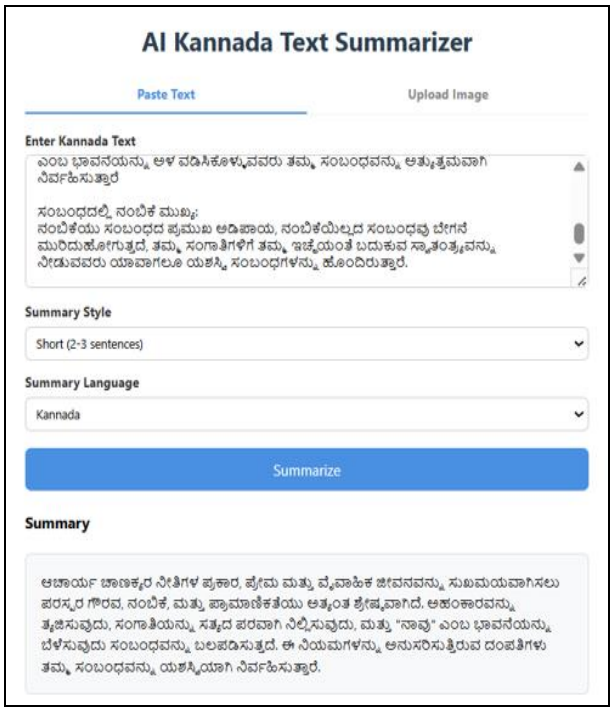

Fig 6 Summary Generated from Digitally Typed Kannada Text.

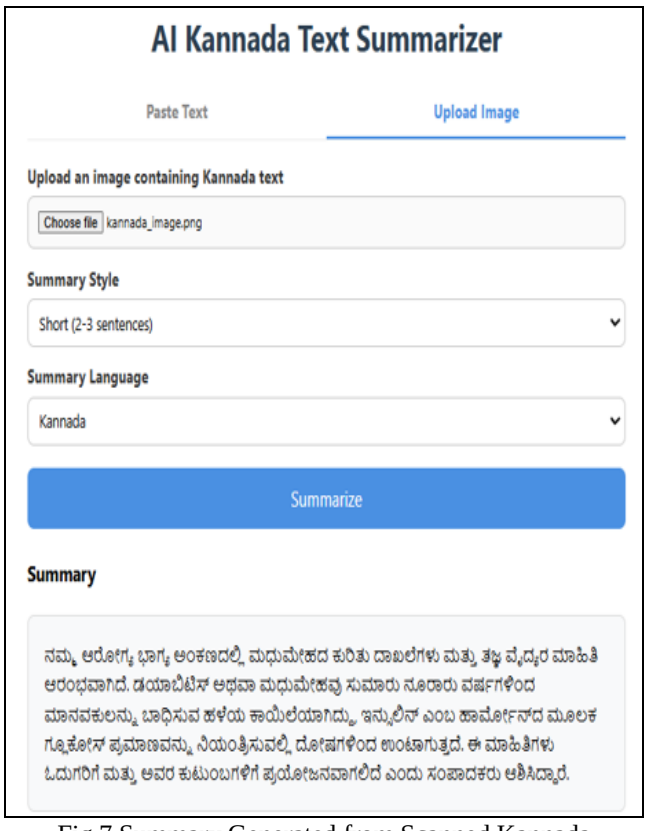

Fig 7 Summary Generated from Scanned Kannada Documents Using OCR.

The text obtained was subjected to normalizing and tokenizing, along with sentence segmentation, removal of stop words, and creation of an extractive summary before presenting it to the user and saving it to the database. The

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26aug453

overall success of this complete process thus proves that the integration of the technologies was done correctly.

The Kannada summary made from typed Kannada text is shown in Fig. 6, and the ability of the application can be seen clearly from here, as it was possible to extract the contents without using OCR process. Likewise, Fig. 7 shows the output from the scanned Kannada document after it has been processed by the OCR process and the summarization.

➢ *Performance Analysis*

The analysis of the newly developed program was carried out by analyzing the operational capabilities of the separate functional modules rather than their algorithmic performance. The OCR module was consistently capable of recognizing the machine-readable Kannada text from scanned images after the images underwent the processing stage. The NLP module was able to perform additional languages processing requiring language-specific preprocessing and summarizing techniques. The interaction among separate components of the system ensured the successful performance of its functions with no interruptions in the process of testing.

From the data in Table 7, which contain the data on how well the OCR and the whole program operated, it is evident that the developed program was able to complete various steps in the document acquisition process, including the OCR processing, the language processing, the summarizing process, and the storing of the information in the database.

Table 7 Module-Wise Functional Performance of the Proposed System

Module	Outcome
Authentication Module	Successfully validated users
OCR Module	Accurate Kannada text extraction
NLP Module	Effective preprocessing
Summarization Module	Generated meaningful summaries
Database Module	Reliable data storage
Result Display Module	Correct summary presentation

➢ *Discussions*

Experimental results reveal the successful integration of optical character recognition (OCR), natural language processing (NLP), and web technologies into a unified system for Kannada document summarization as per the proposed system. Unlike traditional OCR systems, which typically stop after extracting text, the designed system is capable of carrying out the next steps of the process such as language processing, extractive summarization, secure document storage and information visualization. Furthermore, the modular system architecture makes it easier for further software development as each processing module can be updated independently from the systems working on other modules [18], [24], [36].

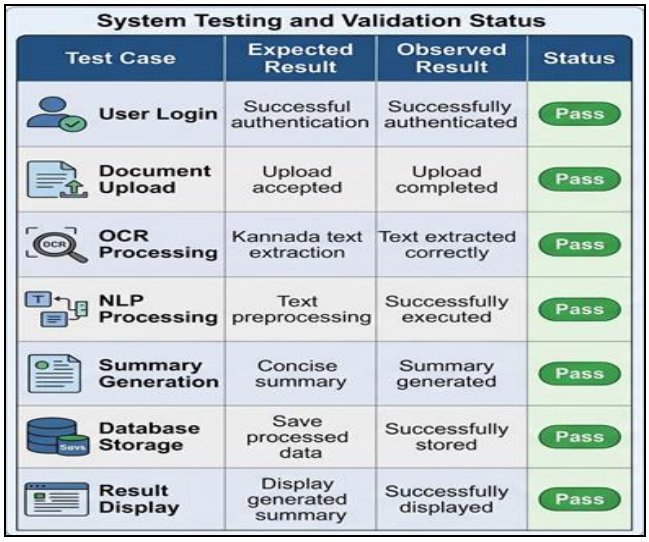

Fig 8 Overall Functional Assessment of the Developed Framework

The software that has been developed can be used especially in those environments where routine processing of large volumes of documents in Kannada takes place, such as universities, governmental institutions, libraries, law firms and research institutions. The fact that the system significantly decreases manual work load necessary for reading Kannada documents means that information contained in them is available for much wider audience of people. Finally, the modular approach used in the system development makes it possible to use different multilingual language processing systems as well as realize OCR of handwritten materials in the next versions of the software. The overall functional assessment of the developed framework is summarized in Fig. 8.

VI. CONCLUSION AND FUTURE RESEARCH DIRECTIONS

With enormous predated kannada physical documents and excessive scanned digital documents, need for translational summarizer is inevitable. The available solution is an Artificial Intelligence (AI) based web framework that integrates the concepts of Optical Character Recognition (OCR), Natural Language Processing (NLP), and extractive text summarization to come up with one entire document processing system. The new system differs from traditional digitalization or summarization systems where digitization of documents and summarization of texts are attempted separately. The proposed AI-based framework is a modular arrangement capable of handling the entire process of production of text data through OCR, language specific pre-processing, generation of summaries, management of databases, and visualization through the web. The developed system can handle both the typed Kannada text and scanned

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26aug453

images/PDFs. The successful performance validation and implementation of all main software components includes the secure user authentication, document upload, image processing; OCR, NI, summarization, DB management, and summarization visualization. The modular software framework allows us to simplify and enhance the maintenance by making it possible to modify individual processing units without affecting the whole system.

The current work has implemented extractive summarization due to efficiency of computation and suitability for non-privileged languages, but this implementation has the possibility of expanding the research in intelligent document understanding in the future. Future developments may involve the usage of state-of-the-art transformer-based language models, like IndicBERT, MuRIL, etc. for the purposes of producing summarization in Kannada language with improved quality of understanding of the context. Future improvements may include the usage of handwriting recognition in Kannada language, multilingual processing of the documents, sentence ranking based on semantic similarity, deployment to the cloud, implementation as mobile applications, and real-time processing of the documents. All the mentioned improvements will enhance the possible application of the suggested framework immensely. To sum up, the proposed solution demonstrates how Artificial Intelligence can be used for regional language document processing through application of OCR, NLP, and web technologies in a single framework. The contribution of the newly developed product is substantial, as it helps to improve the access to Kannada language texts.

REFERENCES

[1]. A. Vaswani et al., "Attention Is All You Need," Advances in Neural Information Processing Systems, vol. 30, pp. 5998–6008, 2017.

[2]. J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," Proc. NAACL-HLT, pp. 4171–4186, 2019.

[3]. C. Raffel et al., "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer," Journal of Machine Learning Research, vol. 21, no. 140, pp. 1–67, 2020.

[4]. M. Lewis et al., "BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation," ACL, pp. 7871–7880, 2020.

[5]. J. Zhang et al., "PEGASUS: Pre-training with Extracted Gap-Sentences for Abstractive Summarization," ICML, pp. 11328–11339, 2020.

[6]. K. Heafield, "KenLM: Faster and Smaller Language Model Queries," Proc. WMT, pp. 187–197, 2011.

[7]. R. Smith, "An Overview of the Tesseract OCR Engine," Proc. ICDAR, pp. 629–633, 2007.

[8]. R. Smith, "History of the Tesseract OCR Engine: What Worked and What Didn't," Document Recognition and Retrieval, vol. 6815, 2009.

[9]. Google, "Tesseract OCR Documentation," 2024.

[10]. G. Bradski, "The OpenCV Library," Dr. Dobb's Journal of Software Tools, 2000.

[11]. A. Rosebrock, Practical Python and OpenCV, PyImageSearch, 2022.

[12]. A. K. Jain, Y. Zhong, and M. Dubuisson-Jolly, "Deformable Template Models: A Review," Signal Processing, vol. 71, no. 2, pp. 109–129, 1998.

[13]. S. Bird, E. Klein, and E. Loper, Natural Language Processing with Python. O'Reilly Media, 2009.

[14]. S. Bird, Natural Language Toolkit (NLTK) Documentation, 2024.

[15]. M. Honnibal and I. Montani, "spaCy 3: Industrial-Strength Natural Language Processing," 2024.

[16]. F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.

[17]. D. Jurafsky and J. H. Martin, Speech and Language Processing, 3rd ed., Pearson, 2024.

[18]. J. Eisenstein, Introduction to Natural Language Processing. MIT Press, 2019.

[19]. R. Mihalcea and P. Tarau, "TextRank: Bringing Order into Text," EMNLP, pp. 404–411, 2004.

[20]. H. P. Luhn, "The Automatic Creation of Literature Abstracts," IBM Journal of Research and Development, vol. 2, no. 2, pp. 159–165, 1958.

[21]. M. Allahyari et al., "Text Summarization Techniques: A Brief Survey," International Journal of Advanced Computer Science and Applications, vol. 8, no. 10, pp. 397–405, 2017.

[22]. Y. Liu and M. Lapata, "Text Summarization with Pretrained Encoders," EMNLP-IJCNLP, pp. 3730–3740, 2019.

[23]. A. K. Singh and B. B. Chaudhuri, "OCR for Indian Scripts: A Survey," Artificial Intelligence Review, vol. 55, pp. 1–31, 2022.

[24]. S. K. Saha et al., "Recent Advances in Optical Character Recognition Using Deep Learning: A Survey," IEEE Access, vol. 10, pp. 110264–110298, 2022.

[25]. P. Choudhary and R. K. Gupta, "Deep Learning-Based OCR for Indian Regional Languages: A Review," Multimedia Tools and Applications, vol. 82, pp. 14231–14262, 2023.

[26]. A. Khanuja et al., "MuRIL: Multilingual Representations for Indian Languages," Findings of ACL, pp. 2658–2669, 2021.

[27]. S. Doddapaneni et al., "IndicBERT: A Multilingual Language Model for Indian Languages," Findings of EMNLP, pp. 5153–5168, 2021.

[28]. R. Kakwani et al., "AI4Bharat IndicNLP Corpus: Monolingual Corpora and Language Models for Indian Languages," ACL Findings, pp. 406–426, 2022.

[29]. K. Krishna, M. S. Reddy, and P. Kumar, "Automatic Text Summarization for Low-Resource Indian Languages: A Review," Journal of King Saud University – Computer and Information Sciences, vol. 35, no. 5, pp. 101675, 2023.

[30]. N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings Using Siamese BERT Networks," EMNLP-IJCNLP, pp. 3982–3992, 2019.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26aug453

[31]. M. Abadi et al., "TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems," Google Research, 2016.
[32]. M. Grinberg, Flask Web Development, 2nd ed. O'Reilly Media, 2018.
[33]. M. Mohri, A. Rostamizadeh, and A. Talwalkar, Foundations of Machine Learning, 3rd ed. MIT Press, 2024.
[34]. S. Saaramsha, "Leveraging NLP for Efficient Kannada Text Summarization," International Journal of Computer Applications, vol. 186, no. 31, 2024.