International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 10, Issue 9, pp 110-116, September-2026

https://doi.org/10.47001/IRJIET/2026.109012

AI-Assisted Letter Validation and Automated Response Generation for Official Letters

[1]Tejas Dange, [2]Vansh Nagpure, [3]Minakshee Chandankhede, [4]S. V. Balamwar

[1,2]Department of Computer Science and Engineering, G H Raisoni College of Engineering and Management, Nagpur, India

[3]Assistant Professor, Department of Computer Science and Engineering, G H Raisoni College of Engineering and Management, Nagpur, India

[4]Associate Scientist, Maharashtra Remote Sensing Application Centre (MRSAC), Nagpur, India

E-mail: [1]tejas.dange.cse@ghrietn.raisoni.net, [2]vansh.nagpure.cse@ghrietn.raisoni.net, [3]minakshee.chandankhede@raisoni.net, [4]sanjay.balamwar@mrsac.maharashtra.gov.in

***Abstract* - Official letters often need to be checked for required information before they can be processed further. This checking is usually done manually and can take time, especially when the letters are received as scanned documents or images. This paper presents an AI-assisted system developed to automate the validation of such letters and generate responses for incomplete submissions. The system accepts letters in PDF, PNG, and JPEG formats and uses Optical Character Recognition (OCR) to extract their text. The extracted text and document details are stored in a database and passed to the validation stage. The implemented format validator checks for required elements such as the date, subject, salutation, sender, and recipient. For content validation, the system is designed to use a domain-specific knowledge base with Retrieval-Augmented Generation (RAG) and a Large Language Model (LLM) to identify missing or incorrect information. The validation findings are then used to generate a response describing the information that needs to be provided or corrected. The system is implemented using Java and Spring Boot with separate components for document processing, OCR, database management, and validation. Synthetic MRSAC-related data is used during development of the AI-assisted components because the actual organizational requirements are confidential.**

Keywords: Letter Validation, OCR, Document Processing, Artificial Intelligence, Automated Response Generation.

I. INTRODUCTION

Official letters and documents are commonly used for communication and data requests in organizations. Before such letters can be processed, they often need to be checked for basic information such as the date, subject, sender, recipient, salutation, and other required details. When this verification is performed manually, checking a large number of documents can take considerable time. The task becomes more difficult when the received documents are scanned copies or images because their contents first have to be converted into text.

Optical Character Recognition (OCR) can be used to extract text from scanned and image-based documents. However, extracting the text does not by itself determine whether a letter contains all the information required for further processing. A letter may have readable text but still be missing important fields or domain-specific information. For this reason, the validation process needs to consider both the structure of the letter and the information contained within it.

This work presents an AI-assisted system for validating official letters and generating responses for incomplete submissions. The system accepts letters in PDF, PNG, and JPEG formats. After upload, the document information and extracted text are stored in a database. The extracted text is then checked using two validation approaches. The first is a format-based validator that checks predefined structural elements. The second is a content-based validation approach that uses a domain-specific knowledge base together with Retrieval-Augmented Generation (RAG) and a Large Language Model (LLM) to examine the information provided in the letter.

The results from the two validation stages can be combined to identify the requirements that have not been satisfied. These findings are then passed to the response-generation component, which is intended to prepare a response explaining the missing information or corrections required from the client. Since actual MRSAC requirements are confidential, synthetic MRSAC-related data is used for developing and validating the AI-assisted components.

The backend of the system is developed using Java and Spring Boot. The application is divided into separate components for document processing, OCR, database interaction, and validation, allowing the different stages of the workflow to be developed and integrated independently.

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 10, Issue 9, pp 110-116, September-2026

https://doi.org/10.47001/IRJIET/2026.109012

II. RELATED WORK

A. Form Understanding and Scanned Document Processing

Abdallah *et al.* reviewed more than 100 studies on form understanding in scanned documents. Their survey discusses the difficulties involved in processing scanned documents, including image noise, complex layouts, and documents containing both printed and handwritten information. The authors identify tasks such as document layout analysis, information extraction, and document visual question answering as important areas in document understanding. They also discuss transformer-based methods that make use of both textual and spatial information when processing scanned documents [1].

The study also points out that extracting the text from a document is not always enough to understand its content. The location and arrangement of information on a page can provide additional meaning, particularly in structured forms [1]. This is relevant to official letter processing because information such as the date, subject, sender, and recipient may appear in specific parts of the document and follow an expected structure.

B. OCR and Generative AI for Document Processing

Kumari *et al.* presented a document-processing approach that combines OCR with Generative AI for text extraction and summarization. Their work discusses the difficulties faced by conventional OCR when dealing with complicated documents and poor-quality scanned copies. In their system, OCR is used to extract the text, after which Google Gemini is used to generate summaries from the extracted content [2].

The workflow described in the study includes document upload, preprocessing, Tesseract-based OCR, and Generative AI-based summarization [2]. The authors show how combining OCR with a generative model can reduce manual effort in document processing. However, their work mainly focuses on extracting and summarizing document content, whereas the system presented in this paper uses the extracted text for letter validation and response generation.

C. LLM-Based Document Understanding

In their work on LayoutLLM, Luo *et al.* explored document understanding by considering the arrangement of information on the page along with the document text [3]. The authors note that directly providing plain or layout text to language models does not fully capture the structural information present in documents. To address this, LayoutLLM uses layout-aware pre-training and supervised fine-tuning, along with a Layout Chain-of-Thought (LayoutCoT) mechanism that helps the model focus on relevant regions of a document [3].

The work demonstrates that document understanding can benefit from considering both textual content and document structure. Its experiments on standard document-understanding benchmarks show improvements over existing approaches based on open-source 7B LLMs/MLLMs [3]. For the proposed letter validation system, this research highlights the importance of structural information when analyzing documents beyond plain OCR text.

D. Retrieval-Augmented Generation for Domain-Specific Knowledge

Brown, Roman, and Devereux conducted a systematic literature review of Retrieval-Augmented Generation (RAG), analyzing 128 studies published between 2020 and May 2025. The review explains that RAG combines the knowledge stored within an LLM with externally retrieved evidence, allowing the model to use domain-specific or updated information during generation [4].

The review identifies a shift from basic retrieval-generation pipelines toward modular RAG systems involving hybrid and structure-aware retrieval, uncertainty-aware processing, memory, and multimodal information. It also highlights retrieval quality, evaluation, efficiency, and security as important considerations when developing dependable RAG systems [4]. These findings support the use of RAG in the proposed system for retrieving relevant domain-specific requirements before performing content-based letter validation.

E. Effect of OCR Quality on RAG Systems

Zhang *et al.* investigated the cascading effect of OCR errors on Retrieval-Augmented Generation systems through the OHRBench benchmark. The authors point out that RAG knowledge bases are frequently constructed from unstructured PDF documents using OCR, making the quality of OCR an important factor in subsequent retrieval and generation performance [5].

The study identifies two major forms of OCR noise: **semantic noise**, caused by recognition errors, and **formatting noise**, caused by incorrect representation of document structures. Experiments using OHRBench showed that OCR errors can negatively affect RAG performance, with semantic noise having a consistent impact while formatting noise affects different retrieval and language-model components differently [5]. This finding is particularly relevant to the proposed system because OCR-extracted letter content is

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 10, Issue 9, pp 110-116, September-2026

https://doi.org/10.47001/IRJIET/2026.109012

subsequently used for domain-specific content validation through RAG.

Research Gap

The reviewed studies demonstrate significant progress in individual components of intelligent document processing. Existing work has explored form understanding and information extraction from scanned documents [1], the integration of OCR with Generative AI for document processing [2], LLM-based understanding of document content and layout [3], and the use of RAG for incorporating external knowledge into LLM-based systems [4]. Furthermore, recent research has shown that OCR quality can directly influence the performance of downstream RAG systems [5].

However, these approaches primarily address individual aspects such as document understanding, information extraction, summarization, or retrieval-based question answering. **The reviewed works do not specifically combine structural letter-format validation, domain-specific content validation, and automated response generation within a single workflow for official letters.** Therefore, this work integrates these complementary techniques into a unified system that first performs conventional format validation, then applies domain-specific RAG and LLM-based analysis for content validation, and finally generates an appropriate response based on the identified requirements.

III. PROPOSED SYSTEM

The proposed system is designed to automate the processing and validation of official letters by combining document processing, Optical Character Recognition (OCR), rule-based format validation, Retrieval-Augmented Generation (RAG), and Large Language Model (LLM)-based analysis. The system is intended to reduce manual effort involved in checking letters and preparing responses for incomplete or incorrect submissions.

A. System Overview

The proposed system is intended to automate the processing and validation of official letters by bringing together document processing, Optical Character Recognition (OCR), rule-based format checking, Retrieval-Augmented Generation (RAG), and Large Language Model (LLM)-based analysis. The main goal is to reduce the manual effort involved in checking submitted letters and preparing responses when required information is missing.

The system accepts official letters in PDF, PNG, or JPEG format. After a document is uploaded, its basic details are stored in the database and the document is sent to the OCR component for text extraction. The extracted text is then stored and made available to the validation modules.

The validation process has two main stages. The first checks whether predefined structural elements, such as the date, subject, salutation, sender, and recipient, are present in the letter. The second stage deals with the actual content of the letter. It uses relevant domain-specific requirements retrieved through a Retrieval-Augmented Generation (RAG) pipeline to provide context for content validation.

The findings from both validation stages are combined to determine which requirements have not been satisfied. These findings are then passed to an LLM-based response-generation component, which prepares a response describing the missing information or corrections required from the client.

B. System Architecture

The overall processing flow of the system is:

Document Upload → OCR/Text Extraction → Database Storage → Format Validation → RAG-Based Knowledge Retrieval → LLM-Based Content Validation → Validation Result → Automated Response Generation

The document upload component serves as the starting point of the workflow. Once a document is received, the backend records its metadata and sends the document to the OCR service. The extracted text is then made available to the validation components.

The format validator works with predefined rules and does not depend on an LLM. For content validation, the system retrieves relevant requirements from a domain-specific knowledge base using RAG. These requirements, together with the extracted letter text, are then provided to an LLM for analysis. The results from the validation stages are subsequently passed to the response-generation component.

C. Document Upload and OCR

The system accepts official letters in PDF, PNG, and JPEG formats. During the upload process, details such as the file name, file type, and other relevant metadata are recorded in the database.

For scanned or image-based documents, OCR is used to convert the visual content into text that can be processed by the application. The extracted text is stored in the database and can then be used by the validation and other processing components.

ISSN (online): 2581-3048
Volume 10, Issue 9, pp 110-116, September-2026
https://doi.org/10.47001/IRJIET/2026.109012

The OCR functionality is kept as a separate service within the backend. This keeps document uploading and text extraction as separate parts of the workflow and allows the OCR component to be changed independently if a different OCR solution is required.

D. Database Storage

The database maintains information related to uploaded documents and their processing. It stores document metadata together with the text extracted through OCR.

The stored information provides a persistent representation of the uploaded document and its extracted content. This allows the validation modules to access the processed text without repeatedly performing OCR on the original document.

E. Letter Validation

Letter validation is divided into **format-based validation** and **content-based validation**.

Format-based validation checks whether required structural elements are present in the letter. Examples include the date, subject, salutation, sender, and recipient. The module identifies missing elements and produces a validation result indicating whether the basic format requirements have been satisfied.

Content-based validation focuses on information that cannot be reliably checked using simple structural rules. The extracted letter content is compared with relevant domain-specific requirements retrieved from the knowledge base through RAG. The retrieved requirements provide contextual information to the LLM, which analyzes the letter and identifies missing or incorrect content.

For development and validation of the AI-assisted components, **synthetic MRSAC-related data** is used as the domain-specific knowledge base because the actual organizational requirements are confidential.

F. Automated Response Generation

After format-based and content-based validations are performed, their findings are combined to determine the issues identified in the submitted letter. These findings are provided to an LLM-based response generation component.

The response generation component produces an appropriate response describing the missing or incorrect information and the corrections required from the client. Thus, instead of only returning a technical validation result, the system aims to provide an understandable response that can assist in the subsequent communication process.

IV. METHODOLOGY

A. Document Processing

The process starts when an official letter is uploaded in PDF, PNG, or JPEG format. The backend receives the file and stores the relevant document information in the database. The uploaded document is then sent to the OCR stage, where its contents are converted into machine-readable text.

The extracted text is used as the input for the validation stages that follow. Keeping document processing separate from validation also allows the extracted text to be reused for different checks without having to process the original document again.

B. OCR Processing

For scanned and image-based letters, OCR is used to convert the visible content of the document into text. The OCR component processes the uploaded file and returns the extracted text to the application.

The extracted text is stored in the database together with the corresponding document information. The format and content validation modules can then use this stored text for further processing.

The quality of the extracted text is important because OCR errors can affect the later stages of the system, especially when the text is used for content analysis and RAG-based retrieval.

C. Format Validation

The first validation stage checks the extracted letter text against a predefined set of structural requirements. The purpose of this stage is to determine whether the basic elements expected in an official letter are present.

The validator checks for elements such as:

- Date
- Subject
- Salutation
- Sender
- Recipient

If one of these required elements is not found in the extracted text, it is added to the list of missing fields. The validation result then contains the overall status along with the fields that were not detected.

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 10, Issue 9, pp 110-116, September-2026

https://doi.org/10.47001/IRJIET/2026.109012

This provides a straightforward rule-based check of the letter structure without requiring an LLM.

D. Content-Based Validation

The format check focuses on the structure of the letter, but it cannot verify whether the information provided in the letter meets the required domain-specific conditions. A separate content-validation stage is therefore used for this purpose.

In this stage, the extracted letter content is analyzed against relevant requirements from a domain-specific knowledge base. Since actual MRSAC requirements are confidential, synthetic MRSAC-related information is used during development and validation of this component.

The relevant requirements are retrieved based on the content of the submitted letter and supplied as contextual information to the LLM. The model then analyzes the letter against these requirements and identifies information that is missing, incomplete, or inconsistent.

E. RAG-Based Knowledge Retrieval

Retrieval-Augmented Generation is used to provide the LLM with relevant domain-specific information during content validation. The domain-specific requirements are stored in a knowledge base and made available for retrieval.

When a letter is submitted for content validation, the relevant portion of the knowledge base is retrieved based on the letter content. The retrieved information is then combined with the extracted letter text and provided to the LLM.

This approach reduces dependence on the model's internally stored knowledge and allows validation to be performed against a defined set of domain requirements. It also allows the knowledge base to be updated independently of the LLM.

F. LLM-Based Analysis and Response Generation

The LLM is used in two stages of the proposed workflow. First, it analyzes the extracted letter content using the requirements retrieved through RAG and identifies content-related issues.

The findings from content validation are then combined with the results of format validation. These combined findings are supplied to the response-generation component, which generates an appropriate response for the submitted letter.

The generated response is intended to clearly communicate the missing or incorrect information and indicate the corrections required from the client. This completes the processing pipeline from document submission to validation and automated response generation.

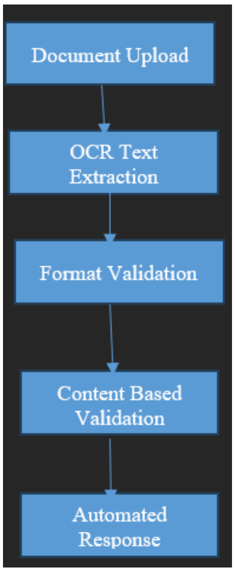

Figure 1: System Flow

V. IMPLEMENTATION

The system is implemented as a Java-based backend using Spring Boot. The current implementation covers document upload, OCR-based text extraction, database storage, and format validation. The backend follows a layered structure so that document handling, processing, and validation are kept in separate components.

A. Backend Architecture

The backend is developed using Java and Spring Boot. The application is organized into controller, service, repository, and exception-handling components. Controllers receive API requests and return the corresponding responses, while the service layer contains the main processing logic. Repository components are responsible for communicating with the database.

Spring Dependency Injection is used to connect the different services. For example, after a document is received, the document service passes it to the OCR service through dependency injection. Keeping these responsibilities separate makes it easier to test and modify individual parts of the application without affecting the rest of the system.

B. Document Upload Service

The document upload service handles the initial submission of a letter to the system. It accepts PDF, PNG, and JPEG files and processes the information associated with the uploaded document.

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 10, Issue 9, pp 110-116, September-2026

https://doi.org/10.47001/IRJIET/2026.109012

When a file is received, its details, including the file name and file type, are stored in the database. The uploaded document is then passed to the OCR service for text extraction. By keeping file handling and OCR processing in separate services, the upload process does not depend directly on the implementation of the OCR component.

C. OCR Service

The OCR functionality is implemented using **Tesseract OCR**. The OCR service receives the uploaded document and extracts the textual information contained within it.

The extracted text is returned to the application and stored in the database along with the corresponding document information. This provides the validation modules with machine-readable content that can be processed without directly interacting with the original scanned document.

The OCR service is implemented as an independent backend service, allowing it to be replaced or enhanced with another OCR solution in the future if required.

D. Validation Service

The validation service processes the OCR-extracted text and performs letter validation.

The currently implemented format validator uses predefined rules to check for required structural elements in an official letter. It identifies missing elements and returns the corresponding validation status and missing fields through the backend API.

The content-validation component is being extended to incorporate RAG and an LLM. The RAG component will retrieve relevant requirements from the synthetic domain-specific knowledge base, after which the LLM will analyze the letter content against the retrieved requirements.

The separation between format and content validation allows deterministic structural checks to operate independently from AI-assisted semantic analysis.

E. Database Integration

Database integration is used to persist document-related information throughout the processing workflow. The database stores uploaded document metadata and the text extracted through OCR.

The backend interacts with the database through repository components, keeping database operations separate from the application and validation logic. The stored information can subsequently be accessed by the validation and response-generation components.

This database-based approach also provides a persistent record of processed documents and their extracted content, supporting further analysis and future extensions of the system.

VI. FUTURE SCOPE

The proposed system can be extended in several directions to improve its accuracy, flexibility, and applicability to real-world official letter processing.

The content-based validation component can be further developed by integrating a larger and more comprehensive domain-specific knowledge base containing organizational requirements, document guidelines, and different categories of official letters. This would allow the RAG pipeline to retrieve more relevant information for different types of submissions.

The system can also be extended to support more advanced document understanding techniques that consider document layout and spatial relationships in addition to extracted text. This can improve processing of complex or inconsistently formatted documents.

Further improvements can be made to the OCR pipeline by incorporating preprocessing techniques and more robust OCR models to improve text extraction from low-quality scans, handwritten content, and complex document layouts. Improving OCR quality can also benefit the downstream RAG and LLM-based validation stages.

The LLM-based validation component can be enhanced to provide more detailed explanations for detected content-related issues and distinguish between missing, incorrect, and inconsistent information. The response-generation component can also be extended to produce standardized responses based on different categories of validation results.

Future versions of the system could additionally incorporate confidence scores, human-in-the-loop verification, multilingual document processing, and support for a wider range of official document types. With access to appropriate organizational data, the system could also be evaluated using real-world documents to measure its performance under operational conditions.

VII. CONCLUSION

This paper presented an AI-assisted system for letter validation and automated response generation for official document processing. The proposed workflow combines document upload, OCR-based text extraction, database

ISSN (online): 2581-3048
Volume 10, Issue 9, pp 110-116, September-2026
https://doi.org/10.47001/IRJIET/2026.109012

storage, format-based validation, Retrieval-Augmented Generation, and Large Language Model-based analysis within a unified processing pipeline.

The implemented prototype demonstrates the initial stages of the system, including document processing, OCR-based text extraction, database storage, and structural letter validation. The implemented format validator checks the required letter fields and reports any fields that are missing.

RAG and LLM-based analysis are being added to handle the content-related checks that cannot be covered by the format validator alone. The automated response generation component further aims to convert validation findings into an understandable response for the client.

The work demonstrates how conventional document-processing techniques can be combined with AI-assisted methods to support automated official letter processing. Although the complete system is still under development, the current implementation establishes the foundation for integrating content validation and automated response generation in future stages.

REFERENCES

[1] A.Abdallah, D. Eberharter, Z. Pfister, and A. Jatowt, “A survey of recent approaches to form understanding in scanned documents,” *Artificial Intelligence Review*, vol. 57, no. 11, Art. no. 342, 2024, doi: 10.1007/s10462-024-11000-0.

[2] C. S. Kumari, V. Y. Gupta, G. Manikanta, and B. V. Abhishikth, “Automated document processing: Combining OCR and generative AI for efficient text extraction and summarization,” *in Proc. 4th Int. Conf. Information Technology, Civil Innovation, Science, and Management (ICITSM 2025)*, 2025, doi: 10.4108/eai.28-4-2025.2357770.

[3] C. Luo, Y. Shen, Z. Zhu, Q. Zheng, Z. Yu, and C. Yao, “LayoutLLM: Layout instruction tuning with large language models for document understanding,” *in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, 2024, pp. 15630–15640.

[4] A.Brown, M. Roman, and B. Devereux, “A systematic literature review of retrieval-augmented generation: Techniques, metrics, and challenges,” *Big Data and Cognitive Computing*, vol. 9, no. 12, Art. no. 320, 2025, doi: 10.3390/bdcc9120320.

[5] J. Zhang, Q. Zhang, B. Wang, L. Ouyang, Z. Wen, Y. Li, K.-H. Chow, C. He, and W. Zhang, “OCR Hinders RAG: Evaluating the cascading impact of OCR on retrieval-augmented generation,” *in Proc. IEEE/CVF Int. Conf. Computer Vision (ICCV)*, 2025, pp. 17443–17453, doi: 10.1109/ICCV51701.2025.01620.

Citation of this Article:

Tejas Dange, Vansh Nagpure, Minakshee Chandankhede, & S. V. Balamwar. (2026). AI-Assisted Letter Validation and Automated Response Generation for Official Letters. *International Research Journal of Innovations in Engineering and Technology - IRJIET*, 10(9), 110-116. Article DOI https://doi.org/10.47001/IRJIET/2026.109012

****

