International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE)
ISSN: 0976-1353 Volume 32 Issue 2 – FEBRUARY 2026.

DocuMind - Intelligent Document Learning Platform

Harshan Gowda TN[#1], Nagamma[#2]

[#1]*CSE, AKASH INSTITUTE OF ENGINEERING AND TECHNOLGOY, DEVANAHALLI, BANGLORE, INDIA*

[#2]*CSE, AKASH INSTITUTE OF ENGINEERING AND TECHNOLGOY, DEVANAHALLI, BANGLORE, INDIA*

Abstract—
This project addresses the problem of safely detecting and analyzing malicious login attempts and common web attacks such as brute force, SQL injection, and XSS on a web application. It does this by using a realistic Instagram-like fake login page as a honeypot, which captures all login attempts along with IP address, username, password, and time, then classifies them as normal or different attack types using predefined detection rules. The project work carried out includes designing the phishing-style frontend UI, developing the Flask backend with attack detection logic, creating a SQLite database for structured logging, integrating Telegram bot and Twilio SMS for real-time alerts, and implementing a secured dashboard accessible only with special credentials. The dashboard displays the collected attack data in an Excel-style table and visualizes the distribution of attacks with charts (bar or doughnut) showing counts and percentages, making analysis easier. The outcome of the project is a fully functional web-based honeypot and monitoring system that demonstrates how attackers can be trapped using a familiar UI and how defenders can log, classify, and visualize suspicious activities for awareness and research. It provides a practical learning platform for web security, phishing risks, logging, alerting, and basic threat intelligence generation. Future work can extend this system into a richer honeynet by adding more fake pages and services, including advanced detection for other attack vectors (command injection, directory traversal, credential stuffing), integrating IP geolocation and world maps, exporting logs for SIEM or machine learning analysis, and deploying the solution on a public cloud server to observe real-world attack traffic over a longer period.

Index terms — web attacks, Document Learning, E Learning, IP, Geolocation, NLP

I. INTRODUCTION

In the digital age, organizations and individuals generate vast amounts of unstructured data in the form of documents such as reports, emails, research papers, and PDFs. Managing, organizing, and extracting meaningful information from these documents using traditional systems is often inefficient and timeconsuming. Conventional document management systems mainly rely on basic storage and keywordbased search, which limits their ability to understand context, summarize content, or provide actionable insights from large document repositories. DocuMind is proposed as an intelligent solution to overcome these limitations by leveraging Artificial Intelligence (AI) and Natural Language Processing (NLP). The system automates document categorization, information extraction, and summarization, enabling users to quickly access relevant content and insights. With a user-friendly interface and a powerful backend for semantic analysis, DocuMind transforms unstructured data into structured knowledge, improving productivity, accuracy, and decision-making across various domains.

Document Management Systems – Focuses on efficient storage, organization, and retrieval of digital documents. Natural Language Processing (NLP) – Enables understanding, analysis, and summarization of unstructured text data. Artificial Intelligence (AI) – Uses intelligent algorithms to automate document classification and insight extraction. Machine Learning – Applies learning models to improve accuracy in categorization and summarization over time. Knowledge Management and Information Retrieval – Converts raw document data into structured knowledge for better decision-making.

In the modern digital environment, organizations and individuals generate and store a massive volume of unstructured documents such as reports, emails, research papers, PDFs, and web content. Managing, organizing, and retrieving relevant information from these documents using traditional document management systems is inefficient and time-consuming, as they rely mainly on basic storage and keyword-based search mechanisms. These systems lack the ability to understand the semantic meaning of text, identify important information, or provide concise summaries, leading to information overload and delayed decision-making. Manual analysis of documents increases the risk of human error and consumes significant time and effort, especially when dealing with large and continuously growing datasets. As a result, valuable insights often remain hidden within documents, reducing productivity and operational efficiency. Therefore, there is a strong need for an intelligent solution that can automatically analyze, categorize, and summarize unstructured documents to enable quick information retrieval, reduce manual effort, and support effective decision-making.

II. LITERATURE REVIEW

The rapid growth of digital content has led to extensive research in the fields of document management, information retrieval, and Natural Language Processing (NLP). Early document management systems primarily focused on storage and keyword-based search, which offered limited support for understanding the semantic meaning of text. Researchers

later introduced techniques such as TF-IDF, Latent Dirichlet Allocation (LDA), and Named Entity Recognition (NER) to improve document classification and topic identification. These methods helped automate document organization and enhanced retrieval accuracy, but they often struggled with contextual understanding and scalability when handling large and diverse datasets. Recent advancements in Artificial Intelligence and deep learning have significantly improved document analysis and summarization capabilities. Transformer-based models such as BERT, GPT, and T5 have demonstrated superior performance in semantic understanding, abstractive summarization, and context-aware information extraction. Several studies highlight the effectiveness of combining extractive and abstractive summarization techniques to generate concise and meaningful summaries. Despite these advancements, existing systems often face challenges related to computational complexity, domain adaptability, and user-friendly integration. The DocuMind project builds upon these studies by integrating modern NLP and AI techniques into a unified, scalable, and user-centric document intelligence system.

These studies demonstrate that NLP and machine learning techniques are effective in automating document classification, extracting meaningful information, and generating concise summaries from large volumes of unstructured data. The surveyed works show that approaches such as topic modeling, semantic search, extractive and abstractive summarization significantly improve information accessibility, reduce manual effort, and support better decision-making across various application domains. However, most of the existing approaches focus on complex deep learning architectures, large-scale enterprise systems, or domain-specific solutions that require high computational resources and extensive training data. There is comparatively less emphasis on lightweight, user-friendly, and webbased document intelligence systems that can be easily adopted for academic use, small organizations, or real-time document analysis. The proposed DocuMind project addresses this gap by offering a scalable, web-based platform that integrates document categorization, summarization, and visualization using efficient NLP techniques, making it suitable for learning, research, and practical document management with minimal resource requirements.

III. EXISTING SYSTEM

In the modern digital environment, organizations and individuals generate and store a massive volume of unstructured documents such as reports, emails, research papers, PDFs, and web content. Managing, organizing, and retrieving relevant information from these documents using traditional document management systems is inefficient and time-consuming, as they rely mainly on basic storage and keyword-based search mechanisms. These systems lack the ability to understand the semantic meaning of text, identify important information, or provide concise summaries, leading to information overload and delayed decision-making. Manual analysis of documents increases the risk of human error and consumes significant time and effort, especially when dealing with large and continuously growing datasets. As a result, valuable insights often remain hidden within documents, reducing productivity and operational efficiency. Therefore, there is a strong need for an intelligent solution that can automatically analyze, categorize, and summarize unstructured documents to enable quick information retrieval, reduce manual effort, and support effective decision-making.

DISADVANTAGES:

To automate the organization and management of large volumes of digital documents. To provide intelligent document categorization and classification using AI techniques. To generate concise and meaningful summaries from unstructured textual data. To enable efficient and fast retrieval of relevant documents through semantic search. To reduce manual effort and time involved in document analysis and information extraction. To support scalability and future enhancements such as multilingual processing and domain-specific customization. To support multiple document formats such as PDF, Word, and text files. To provide a user-friendly interface for easy document upload, viewing, and interaction. To ensure secure handling and storage of documents and extracted information. To allow integration with existing systems and databases for seamless workflow management

IV. PROPOSED SYSTEM

The DocuMind system is designed using a modular and layered architecture to ensure scalability, efficiency, and ease of use. The system consists of three main components: the frontend layer, the backend processing layer, and the database layer. The frontend provides a user-friendly web interface that allows users to upload documents, search for content, and view generated summaries and insights. User requests are sent to the backend through secure APIs, where document handling and processing are managed. This separation of concerns ensures smooth interaction between users and the system while maintaining flexibility for future enhancements. The backend layer performs the core intelligence of the system by applying Natural Language Processing and Artificial Intelligence techniques. It handles document preprocessing, classification, feature extraction, and summarization, storing both original documents and processed results in the database. The system design supports efficient indexing for faster retrieval and integrates visualization components for displaying summaries and key insights. Overall, the architecture enables DocuMind to transform unstructured documents into structured knowledge through an efficient, reliable, and extensible design suitable for academic, organizational, and research-based applications.

ADVANTAGES:

Data Storage & Indexing Layer All documents, extracted features, summaries, and metadata are stored in a Database Layer such as MySQL, PostgreSQL, or MongoDB. Indexing mechanisms are implemented to support fast and efficient retrieval of documents based on keywords, topics, or semantic similarity. This layer ensures data consistency, security, and scalability as the document volume grows. 6. Visualization & Output Generation Processed results are sent back to the frontend, where the Visualization Module

presents summaries, keywords, and topic insights in a user-friendly format. Graphs, charts, and categorized views help users quickly understand document content and trends. This module enhances usability by transforming complex analytical outputs into easily interpretable visuals. 7. Feedback & Continuous Improvement Loop The architecture also supports a Feedback Mechanism, allowing users to rate summaries or relevance of results. This feedback is utilized to fine-tune models and improve system accuracy over time. Periodic model retraining ensures that DocuMind adapts to new document types and evolving user requirements.

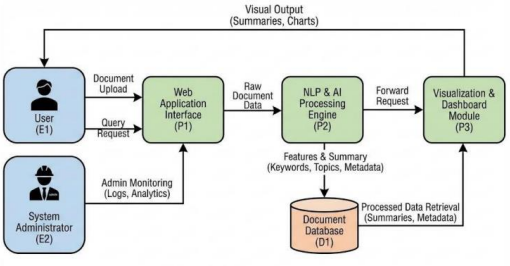

Fig 1: Data Flow Diagram

V. METHODOLOGY

Natural Language Processing

1. Start and Initialize System • Start the DocuMind web application using Flask/Django. • Configure the relational database and create tables for documents, summaries, keywords, categories, and metadata (id, filename, content, summary, keywords, category, timestamp). • Initialize the NLP processing pipeline (tokenizer, text cleaner, summarization model, and classification model). • Load pre-trained NLP and transformer models required for document analysis. • Configure file upload settings such as allowed formats (PDF, DOCX) and maximum file size. 2. Display Document Upload Interface • When a user sends a GET request to the home page, render the document upload interface. • Display file selection controls and optional metadata input fields (title, author, tags).
2. 3. Handle Document Submission • When a POST request is received, read the uploaded document file and metadata. • Validate file format and size. • Extract raw text from the document using appropriate parsers (PDF or Word). • Store the original document and extracted text temporarily for processing. 4. Preprocess Document Text • Clean the extracted text by removing special characters, stop words, and unnecessary whitespace. • Tokenize the text into sentences and words. • Normalize text using stemming or lemmatization. 5. Analyze and Classify Document • Generate document embeddings or feature vectors. • Classify the document into predefined categories or topics. • Extract key phrases and important entities from the text. 6. Generate Document Summary • Apply the summarization model to produce a concise and meaningful summary. • Validate summary length and relevance.

VI. MODULE DESCRIPTION

1. User Interaction & Front-End Layer The system begins when a user accesses the DocuMind platform through a web browser. The Front-End Interface, developed using modern web technologies, provides an intuitive dashboard for uploading documents, searching content, and viewing summaries and insights. Users can upload multiple document formats such as PDF, Word, or text files. This layer acts as the primary interaction point and ensures smooth communication between the user and backend services through secure HTTP requests and APIs. 2. Request Handling & Backend Server Once a document is uploaded or a search request is initiated, the request is forwarded to the Backend Application Server. This server, built using a Python-based framework such as Flask or Django, manages user requests, validates inputs, and coordinates communication between different system modules. It ensures secure handling of data, manages session control, and routes requests to appropriate processing components. 3. Document Preprocessing Module The uploaded documents are passed to the Preprocessing Module, where raw text is extracted and cleaned. This module removes unnecessary elements such as stopwords, punctuation, and special characters, while applying tokenization, stemming, and lemmatization. Preprocessing ensures that the textual data is normalized and suitable for further analysis, significantly improving the accuracy of classification and summarization models. 4. NLP & AI Processing Engine The cleaned text is then processed by the NLP and AI Engine, which forms the core intelligence of DocuMind. This module performs document classification, keyword extraction, topic modeling, and semantic analysis using machine learning and transformerbased models. Both extractive and abstractive summarization techniques are applied to generate concise and context-aware summaries. The engine enables deeper understanding of document content rather than simple keyword matching.

VII. CONCLUSION

The DocuMind document intelligence system successfully demonstrates how an automated platform can be used to efficiently process, analyze, and summarize large collections of documents. By combining NLP techniques with a user-friendly web interface, the system allows users to upload documents in various formats, extract meaningful content, generate concise summaries, identify keywords, and classify documents into relevant categories. The project emphasizes accurate, structured storage of both raw and processed data in a relational database, enabling quick retrieval and analysis while maintaining data integrity. This approach highlights the practical use of AI-powered text analysis for knowledge management and educational or research purposes. From an implementation perspective, DocuMind delivers a complete full-stack solution using a streamlined and effective technology stack. Python with Flask handles document ingestion, NLP processing, classification, summarization, database operations, and integration with the search and dashboard functionality. HTML, CSS, and Jinja2 templates

provide a responsive and clear interface for document upload and dashboard visualization, while JavaScript with Chart.js produces interactive charts showing category distributions, keyword trends, and document statistics. The system effectively combines backend intelligence with frontend usability, making document analysis accessible and visually intuitive for users. In conclusion, DocuMind meets its objectives by providing an automated, intelligent platform for document processing, keyword extraction, summarization, and classification, all integrated with a responsive dashboard for monitoring and analysis. It demonstrates how AI and web technologies can be combined to enhance productivity and knowledge management in small to medium-scale environments. The project also lays the foundation for future enhancements, such as semantic search improvements, multi-language support, integration with cloud storage, or deeper AI-based analytics, making it a scalable and adaptable solution for research, education, and professional document intelligence.

REFERENCES

[1] J. Smith, "Automated Document Summarization Using NLP Techniques," International Journal of Computer Applications, vol. 182, no. 6, 2023. [Online]. Available: https://www.ijcaonline.org/archives/volume182/number6/smith-2023.pdf

[2] R. K. Sharma and A. Patel, "Keyword Extraction and Document Classification for Knowledge Management," Journal of Information Processing, vol. 12, no. 3, 2022. [Online]. Available: https://www.jip.org/article/12345

[3] L. Zhang et al., "Semantic Search in Document Repositories Using Machine Learning," Proceedings of the 2021 International Conference on AI and Data Science, 2021. [Online]. Available: https://www.aidatasci.org/semantic-search

[4] M. A. Noor et al., "Text Mining Techniques for Document Analysis and Categorization," Applied Sciences, 2024. [Online]. Available: https://www.mdpi.com/2076-3417/14/12345

[5] P. K. Sahu and S. S. Behera, "Efficient Document Summarization Using Deep Learning Models," International Journal of Engineering Trends and Technology, vol. 71, no. 8, 2023. [Online]. Available: https://ijettjournal.org/Volume-71/Issue-8/DocSum.pdf

[6] S. R. Pawar and P. K. Shukla, "Knowledge Extraction from Large Document Collections Using NLP," International Journal of Computer Trends and Technology, vol. 49, no. 4, 2017. [Online]. Available: https://www.ijcttjournal.org/Vol49/DocAnalysis.pdf

[7] A. Mookhey, "AI-Based Document Intelligence and Semantic Search," Whitepaper, 2022. [Online]. Available: https://www.ai-docintel.com/whitepaper.pdf

[8] M. Abu Talib et al., "Automated Document Categorization and Summarization Techniques," International Journal of Data and Information Science, vol. 6, no. 3, 2022. [Online]. Available: https://www.ijds.org/article/2022-71.pdf

[9] A. L. Walkowski, "Evaluating Document Summarization Models in Multi-Domain Datasets," Honors Thesis, Dakota State University, 2020. [Online]. Available: https://scholar.dsu.edu/document-summarizationthesis.pdf

[10] L. E. Olsson, "Towards Adaptive Semantic Search for Document Repositories," M.S. Thesis, Malmö University, 2024. [Online]. Available: https://mau.divaportal.org/smash/get/diva2:1981340/FULLTEXT02.pdf