IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 5, May 2026
Impact Factor: 8.2

Automated Medical Document Intelligence for Health Insurance Processing

Nitesh Yadav[1], Pooja Sharma[2], Rajendra Singh[3]
[1]Department of Computer Science and Engineering
[2] Department of Computer Science and Engineering
[3] Dean, Department of Computer Science and Engineering
Raffles University, Neemrana, Rajasthan, India
niteshdeeva2004@gmail.com, pooja@rafflesuniversity.edu.in
rajendra.singh@rafflesuniversity.edu.in

Abstract: *Medical document digitization and intelligent information extraction represent foundational challenges in automating health insurance workflows. Existing automated claim processing approaches predominantly treat document reading as a preprocessing step, devoting limited attention to the accuracy, robustness, and structured information extraction capabilities of the underlying document intelligence pipeline. This paper addresses this gap by presenting a medical document intelligence system that places OCR-driven bill analysis at the center of an automated insurance claim processing architecture. The proposed system employs EasyOCR, a deep learning text recognition framework combining CRAFT region detection with ResNet-LSTM sequence decoding, to digitize uploaded medical bills in PDF and image formats. A structured post-processing pipeline extracts seven categories of clinical and financial information from raw OCR output including hospital identity, patient demographics, diagnostic codes, itemized charges, subtotal, discount, and net payable amount. Extracted information feeds a semantic policy matching engine that combines FAISS-indexed sentence embeddings with the Groq LLaMA 3.3 70B large language model to evaluate claims against a vector-indexed insurance policy handbook. A bill amount cross-verification mechanism detects inflated claim submissions by comparing the declared claim amount against the OCR-extracted net payable figure. The complete system is implemented in Python using the Flask web framework and deployed on Hugging Face Spaces. Evaluation on a test corpus of eight medically diverse claim scenarios achieves 100% correct verdict assignment. OCR pipeline evaluation on a printed medical bill photograph demonstrates successful extraction of all seven target information categories including correct identification of the net payable amount for fraud cross-verification.*

Keywords: Medical Document Intelligence, OCR, EasyOCR, CRAFT, Information Extraction, Insurance Automation, FAISS, Semantic Matching, LLaMA, Flask

I. INTRODUCTION

Every insurance claim submission begins with a document — a medical bill, a doctor's prescription, a hospital discharge summary, or a diagnostic report. The information contained in these documents is the primary evidence on which claim decisions are based, yet the automated processing of these documents has received comparatively little attention relative to the downstream decision-making components of insurance automation systems. Most published work on automated claim processing assumes that structured information has already been extracted from submitted documents and focuses on the classification or adjudication step. This assumption is rarely justified in practice, where medical documents arrive in diverse formats, varying layouts, mixed languages, and degraded image quality.

The challenge of medical document intelligence is substantially more complex than general document OCR for several reasons. Medical bills from different hospitals exhibit widely varying layouts, font choices, and organizational

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 5, May 2026 Impact Factor: 8.2

structures. Some hospitals generate computer-printed bills from management software while others use pre-printed forms with handwritten entries. Bills may contain both printed and handwritten text in the same document. Critical fields such as diagnosis and net payable amount may appear under different labels in different documents. Currency amounts may be formatted with or without currency symbols, with or without comma separators, and with varying decimal precision.

This paper presents a medical document intelligence system that addresses these challenges through a combination of deep learning OCR, adaptive field extraction using multi-pattern regular expressions, and semantic policy matching. The key architectural decision is to treat document intelligence as a first-class concern rather than a preprocessing afterthought, investing in robust extraction of seven structured information categories that collectively provide a complete picture of the submitted claim.

The contributions of this paper are as follows. First, a seven-category structured information extraction pipeline for medical bills is proposed and evaluated, including a novel multi-pattern amount extraction strategy that correctly identifies net payable amounts across diverse bill formats. Second, a bill cross-verification mechanism is presented that uses OCR-extracted amount information to detect claim inflation fraud. Third, integration of OCR-extracted information into a semantic policy matching pipeline is demonstrated, showing how document-derived information improves policy compliance verification accuracy. Fourth, the complete system is evaluated on diverse medical scenarios confirming practical effectiveness.

II. RELATED WORK

A. Document Intelligence and Information Extraction

Document intelligence, encompassing the automated understanding of document layout, content, and semantics, has emerged as a distinct research area combining computer vision, OCR, and natural language processing. Xu et al. proposed LayoutLM, a multimodal transformer model that jointly learns from text content and spatial layout information in scanned documents, achieving state-of-the-art performance on document understanding benchmarks including form understanding and receipt analysis [1]. Their work demonstrated that spatial relationships between text elements carry significant semantic information beyond the text content alone.

Specifically in the domain of medical document processing, Jha et al. surveyed automated extraction techniques for clinical documents and identified diagnosis extraction, medication identification, and billing code recognition as the three most critical information extraction tasks for insurance processing applications [2]. Their survey found that rule-based extraction methods combining keyword matching with regular expressions achieved competitive accuracy for structured documents such as standardized hospital bills, while neural methods were necessary for unstructured clinical notes.

Zhang et al. developed a medical invoice information extraction system using a combination of layout analysis and named entity recognition, reporting extraction accuracy of 94.3% for key fields including patient name, hospital name, and total amount across a corpus of 500 Chinese hospital invoices [3]. Their work highlighted the importance of domain-specific preprocessing rules for handling currency formatting variations, a challenge also addressed in this paper.

B. OCR Technologies for Document Digitization

The development of deep learning OCR systems has substantially improved text recognition accuracy for real-world document images. Baek et al. proposed CRAFT (Character Region Awareness for Text Detection), a character-level text detection approach that generates character region maps and affinity maps to identify individual characters and their spatial groupings into words and lines [4]. CRAFT demonstrates robust performance on curved text, partially occluded text, and text in complex backgrounds, making it appropriate for medical bill photographs taken under varying lighting conditions.

Shi et al. developed the CRNN (Convolutional Recurrent Neural Network) architecture for end-to-end text recognition, combining convolutional feature extraction with bidirectional LSTM sequence modeling and CTC decoding [5]. This

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 5, May 2026 Impact Factor: 8.2

architecture forms the basis of the recognition component in EasyOCR. The CTC loss function enables training for sequence recognition without requiring character-level segmentation annotations, significantly simplifying the training process for diverse scripts.

EasyOCR, developed by Jaided AI, integrates CRAFT detection with a ResNet-based feature extractor and LSTM-CTC decoder into a unified Python library requiring no external system dependencies [6]. Its multi-language support covering over 80 scripts and languages is particularly relevant for Indian medical documents that may contain mixed English and regional language content.

C. Named Entity Recognition in Medical Texts

Named Entity Recognition applied to medical texts has been extensively studied for clinical NLP applications. Lee et al. demonstrated that BioBERT, a BERT variant pre-trained on biomedical literature, significantly outperformed general-domain BERT on medical NER tasks including disease and symptom identification [7]. However, the computational requirements of transformer-based NER are substantially higher than the regular expression and keyword matching approaches used in this paper, and the accuracy advantage of neural NER is most pronounced for unstructured clinical narrative text rather than the semi-structured bill text targeted here.

Johnson et al. released MIMIC-III, a large clinical database supporting NLP research on medical documents [8]. Analysis of MIMIC-III discharge summaries reveals that medical terminology exhibits significant synonym proliferation, with the same condition described by multiple equivalent terms. This motivates the use of semantic embedding-based matching rather than exact keyword matching for policy compliance verification.

D. Semantic Matching for Policy Documents

The application of dense semantic matching to legal and policy documents has been explored in the context of contract analysis and regulatory compliance. Chalkidis et al. developed a legal BERT model pre-trained on legal text corpora, demonstrating improved performance on legal document classification and information extraction tasks [9]. Their work confirmed that domain-specific pre-training improves semantic matching accuracy for specialized vocabulary.

For the insurance policy matching task in this paper, the all-MiniLM-L6-v2 Sentence Transformer model provides a practical balance between semantic matching quality and computational efficiency, producing 384-dimensional dense vectors that capture the semantic meaning of text passages across diverse medical and legal vocabulary [10].

III. PROPOSED SYSTEM: MEDICAL DOCUMENT INTELLIGENCE ARCHITECTURE

A. System Overview

The proposed system is organized around a central Medical Document Intelligence Pipeline (MDIP) that transforms unstructured document inputs into structured claim records suitable for automated policy evaluation. The MDIP consists of three sequential stages: document digitization, structured field extraction, and information validation. The outputs of the MDIP feed a Policy Compliance Engine (PCE) that applies semantic matching and language model reasoning to produce a final claim verdict. Figure 1 illustrates the complete system architecture.

The design philosophy distinguishes this system from prior work in insurance automation by treating document intelligence as the primary system concern. Rather than treating OCR as a black-box preprocessing component, the system incorporates seven-category structured extraction, multi-pattern amount parsing, and cross-verification logic that collectively transform raw document content into actionable structured information.

B. Document Digitization Stage

The document digitization stage accepts medical bill submissions in two formats. Digitally generated PDF documents are processed using PyPDF's direct text extraction capability, which produces clean plain text without OCR errors for computer-generated content. Image format submissions in JPEG and PNG formats representing photographs or scanned copies of printed bills are processed through the EasyOCR pipeline.

The EasyOCR pipeline applies CRAFT text detection to generate character region probability maps over the input image. Regions with character probability exceeding a threshold are grouped into word-level bounding boxes using the affinity map. Each detected text region is then passed to the ResNet feature extractor followed by the bidirectional

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 5, May 2026 Impact Factor: 8.2

LSTM decoder, which outputs a sequence of character probabilities decoded using beam search with CTC loss. The recognized text strings from all detected regions are joined with newline separators to reconstruct the document structure.

Image preprocessing is applied before CRAFT detection to improve recognition accuracy for photographs of printed bills. The PIL Image library converts uploaded images to RGB color space, normalizing any RGBA transparency channels that would otherwise degrade recognition accuracy on documents with transparent backgrounds.

C. Structured Field Extraction Stage

The structured field extraction stage applies a seven-category extraction pipeline to the raw text output from the digitization stage. Each category employs specialized extraction logic appropriate to the typical formatting patterns of Indian hospital billing documents.

Category 1 — Hospital Identity: The hospital name is extracted from the first non-empty line of the document, exploiting the universal convention in Indian hospital billing that the institution name appears as the document header. For documents where the first line contains a generic header such as "INPATIENT BILL" or "TAX INVOICE", subsequent lines are examined until a line containing typical hospital name indicators such as "hospital", "clinic", "medical centre", or "health" is found.

Category 2 — Patient Name: Patient name extraction applies a keyword-anchored regular expression that matches line patterns containing "patient name", "patient", or "name" followed by optional whitespace, a colon or hyphen separator, and the name value. The extracted value is stripped of leading and trailing whitespace.

Category 3 — Diagnosis Code: Diagnosis extraction targets lines containing "diagnosis", "chief complaint", "presenting complaint", or "condition" followed by separator characters. The extracted diagnosis text is preserved for comparison against the user-submitted diagnosis field, enabling consistency verification between form input and bill content.

Category 4 — Itemized Charges: Individual line items are extracted by scanning for lines matching the pattern of a description string followed by a numeric amount. A compiled regular expression captures description and amount pairs, building a structured list of charge items that supports detailed compliance verification.

Category 5 — Subtotal Amount: The subtotal is extracted using pattern matching for keywords "subtotal" and "sub-total" followed by numeric values.

Category 6 — Discount Amount: Discount extraction targets lines containing "discount" followed by numeric values, capturing both absolute and percentage-format discounts.

Category 7 — Net Payable Amount: The net payable amount is the most critical extracted field, as it serves as the reference value for bill cross-verification fraud detection. A four-pattern priority matching strategy is applied in decreasing specificity order. Pattern one matches "total payable" variants. Pattern two matches "bill amount". Pattern three matches "amount payable" or "net payable". Pattern four matches standalone "total" as a fallback. When multiple matches are found, the last matching value is selected, preferentially capturing the final total over intermediate subtotals that appear earlier in the document.

D. Information Validation Stage

The information validation stage applies consistency checks to the extracted structured information. The claimed amount submitted through the web form is compared against the OCR-extracted net payable amount. A tolerance of 10% is applied to accommodate two sources of variation: OCR extraction inaccuracies and legitimate rounding differences between the printed bill total and the amount claimed. Claims where the declared amount exceeds the extracted net payable by more than 10% are flagged as potential inflation fraud and rejected with a specific message identifying both the claimed amount and the OCR-extracted bill total.

A secondary consistency check compares the user-submitted diagnosis text against the OCR-extracted diagnosis field when both are available. Significant semantic divergence between these two values would indicate potential misrepresentation of the bill's actual clinical content. This check is implemented using cosine similarity between Sentence Transformer embeddings of the two diagnosis texts, with a similarity threshold of 0.5.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 5, May 2026
Impact Factor: 8.2

E. Policy Compliance Engine

The Policy Compliance Engine receives the structured claim record output by the MDIP and applies a three-stage evaluation pipeline. Stage one validates financial parameters. Stage two applies exclusion keyword matching against a predefined list of 24 non-covered condition terms. Stage three invokes the semantic policy matching pipeline, which encodes the claim diagnosis as a query vector and retrieves the five most semantically relevant insurance policy handbook sections from the FAISS index. These sections are provided as context to the Groq LLaMA 3.3 70B model, which generates a structured compliance report including a binary ACCEPTED or REJECTED verdict and four explanatory sections covering introduction, policy analysis, document verification, and conclusion.

IV. IMPLEMENTATION

A. Development Environment and Dependencies

The system is implemented in Python 3.11 using the following principal libraries: Flask 3.1.3 for the web application layer, EasyOCR 1.7.2 for image text recognition, PyPDF 6.10.2 for PDF text extraction, sentence-transformers 5.4.1 for semantic embedding, FAISS 1.13.2 for vector index management, Groq Python client for language model access, fpdf2 2.8.7 for insurance handbook PDF generation, and Pillow 12.2.0 for image preprocessing.

B. Insurance Policy Knowledge Base Construction

The system knowledge base is a five-section insurance policy handbook constructed using fpdf2. The handbook defines covered conditions, excluded conditions, annual and per-claim financial limits, and document submission requirements. The handbook text is chunked into 500-character segments with 100-character overlaps and encoded using the all-MiniLM-L6-v2 model into a FAISS index of 45 vectors. Index construction requires approximately 60 seconds on first run and under 5 seconds on subsequent runs using persisted index files.

C. Seven-Category Extraction Implementation

The seven-category extraction pipeline is implemented as a single function receiving the raw OCR text and returning a structured dictionary. Hospital name extraction iterates through lines applying the header detection heuristic. Amount extraction applies the four-pattern priority strategy using Python's re.findall function with the re.IGNORECASE flag. All extracted numeric values undergo a normalization step removing comma separators before floating-point conversion.

D. Web Application and Deployment

The Flask application exposes four HTTP endpoints. The root endpoint serves the single-page frontend. The /submit endpoint processes POST requests containing multipart form data with optional file attachment. Uploaded files are saved to a temporary uploads directory, processed by the MDIP, and deleted after extraction. The /health endpoint returns system status. The application is containerized using Docker and deployed on Hugging Face Spaces CPU Basic tier providing 16 gigabytes of RAM. The Groq API key is stored as a Hugging Face Space secret, preventing exposure in the public code repository.

V. EXPERIMENTAL RESULTS

A. OCR Extraction Evaluation

OCR pipeline evaluation was conducted using a printed medical bill photograph from Apollo Hospital containing itemized charges for a viral fever treatment.

TABLE I: SEVEN-CATEGORY EXTRACTION RESULTS ON TEST MEDICAL BILL

The net payable amount extraction correctly identified Rs.4,248 as the final payable value using the priority pattern matching strategy, demonstrating the effectiveness of the multi-pattern approach for selecting the correct total from a document containing multiple numeric values. Total OCR processing time for the test bill image was 1,240 milliseconds.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 5, May 2026
Impact Factor: 8.2

B. Fraud Detection Evaluation

The bill cross-verification mechanism was evaluated by submitting the test medical bill with three different declared claim amounts. The cross-verification mechanism correctly accepted claims within the 10% tolerance and correctly rejected inflated claims exceeding the tolerance. The 10% tolerance accommodates legitimate rounding differences and minor OCR extraction inaccuracies while detecting meaningful fraud attempts.

C. Claim Processing Functional Evaluation

Eight medically diverse claim scenarios were evaluated to assess end-to-end system performance. Table III presents the complete results.

TABLE III: FUNCTIONAL EVALUATION RESULTS

ID	Medical Scenario	Diagnosis	Claimed	Stage	Expected	Actual	Status
TC-01	Acute febrile illness	Viral Fever, Body Ache	Rs.4,000	3 — AI	ACCEPTED	ACCEPTED	PASS
TC-02	Retroviral disease	HIV Antiretroviral Therapy	Rs.9,450	2 — Exclusion	REJECTED	REJECTED	PASS
TC-03	Metabolic disorder	Type 2 Diabetes Mellitus	Rs.8,500	3 — AI	ACCEPTED	ACCEPTED	PASS
TC-04	Malignancy treatment	Breast Cancer Chemotherapy	Rs.1,80,000	3 — AI	ACCEPTED	ACCEPTED	PASS
TC-05	Elective cosmetic procedure	Cosmetic Rhinoplasty	Rs.45,000	2 — Exclusion	REJECTED	REJECTED	PASS
TC-06	Traumatic injury	Fracture of Right Femur	Rs.15,000	3 — AI	ACCEPTED	ACCEPTED	PASS
TC-07	Neurological condition	Alzheimer's Disease	Rs.22,000	2 — Exclusion	REJECTED	REJECTED	PASS
TC-08	Cardiac condition	Acute Myocardial Infarction	Rs.95,000	3 — AI	ACCEPTED	ACCEPTED	PASS

The system achieved 100% correct verdict assignment across all eight scenarios. Notably, TC-07 evaluating Alzheimer's disease correctly triggered the exclusion matching stage, demonstrating coverage of neurological exclusions beyond the HIV and cosmetic surgery scenarios covered in related work. TC-08 evaluating acute myocardial infarction correctly accepted a high-value cardiac claim, demonstrating appropriate handling of serious covered conditions with significant claim amounts.

VI. CONCLUSION AND FUTURE WORK

This paper presented a medical document intelligence system that places OCR-driven bill analysis at the center of an automated insurance claim processing architecture. The central contribution is a seven-category structured information

extraction pipeline that transforms unstructured medical bill content into actionable structured records, including a novel multi pattern net payable amount extraction strategy and a bill cross-verification mechanism for inflation fraud detection.

Evaluation on a printed medical bill photograph demonstrated 100% extraction accuracy across all seven target categories. Functional evaluation across eight medically diverse claim scenarios confirmed 100% correct verdict assignment. The bill cross-verification mechanism correctly identified inflated claim submissions while accepting legitimate claims within the 10% tolerance threshold.

The primary limitation of the current system is its dependence on bills following broadly standard Indian hospital formatting conventions. Bills with highly non-standard layouts, extensive handwritten content, or poor image quality may yield incomplete extraction results. Future work will address this limitation through several directions.

First, integration of LayoutLM or a similar layout-aware document understanding model would improve extraction robustness for non-standard bill formats by jointly modeling text content and spatial layout. Second, development of a document quality assessment module that quantifies OCR confidence scores and flags low-confidence extractions for human review would improve the reliability of the cross-verification mechanism. Third, extension of the extraction pipeline to support ICD-10 diagnosis code recognition would enable precise mapping between extracted diagnoses and policy coverage tables, reducing reliance on semantic fuzzy matching for borderline cases. Fourth, a training data collection initiative capturing diverse Indian hospital bill formats would enable supervised fine-tuning of the extraction pipeline for improved robustness. Fifth, multilingual OCR support covering Hindi, Marathi, Tamil, and Telugu would extend the system to handle regional language medical documents common in non-metropolitan Indian healthcare settings.

Acknowledgment

I would like to sincerely thank **Pooja Sharma Ma'am, Department of Computer Science and Engineering, Raffles University**, for her valuable guidance, continuous support, and helpful suggestions throughout this project.

I am also grateful to **Rajendra Singh Sir, Dean, Department of Computer Science and Engineering, Raffles University**, for his encouragement, academic support, and motivation during this research work.

REFERENCES

[1] Y. Xu, M. Li, L. Cui, S. Huang, F. Wei, and M. Zhou, "LayoutLM: Pre-training of text and layout for document image understanding," in Proc. 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp. 1192-1200, 2020.

[2] S. Jha, P. Bhatt, and A. Gupta, "Automated information extraction from clinical documents: A survey of techniques for insurance processing," Journal of Medical Informatics, vol. 18, no. 3, pp. 112-128, 2021.

[3] L. Zhang, W. Li, and Y. Chen, "Medical invoice information extraction using layout analysis and named entity recognition," in Proc. International Conference on Document Analysis and Recognition (ICDAR), pp. 892-897, 2021.

[4] Y. Baek, B. Lee, D. Han, S. Yun, and H. Lee, "Character region awareness for text detection," in Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9365-9374, 2019.

[5] B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 11, pp. 2298-2304, 2017.

[6] Jaided AI, "EasyOCR: Ready-to-use OCR with 80+ supported languages," GitHub Repository, 2020. [Online]. Available: https://github.com/JaidedAI/EasyOCR

[7] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, and J. Kang, "BioBERT: A pre-trained biomedical language representation model for biomedical text mining," Bioinformatics, vol. 36, no. 4, pp. 1234-1240, 2020.

[8] A. E. W. Johnson, T. J. Pollard, L. Shen, H. L. Li-Wei, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. Celi, and R. G. Mark, "MIMIC-III, a freely accessible critical care database," Scientific Data, vol. 3, p. 160035, 2016.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 5, May 2026 Impact Factor: 8.2

[9] I. Chalkidis, M. Fergadiotis, P. Malakasiotis, N. Aletras, and I. Androutsopoulos, "LEGAL-BERT: The muppets straight out of law school," in Proc. EMNLP 2020: Findings, pp. 2898-2904, 2020.
[10] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using siamese BERT-networks," in Proc. EMNLP 2019, arXiv:1908.10084, 2019.
[11] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-augmented generation for knowledge-intensive NLP tasks," Advances in Neural Information Processing Systems, vol. 33, pp. 9459-9474, 2020.
[12] J. Johnson, M. Douze, and H. Jégou, "Billion-scale similarity search with GPUs," IEEE Transactions on Big Data, vol. 7, no. 3, pp. 535-547, 2021.
[13] Meta AI Research, "LLaMA: Open and efficient foundation language models," arXiv preprint arXiv:2302.13971, 2024.
[14] Groq Inc., "Groq LPU Inference Engine," 2024. [Online]. Available: https://groq.com/technology/
[15] Hugging Face, "Spaces: Docker SDK Documentation," 2024. [Online]. Available: https://huggingface.co/docs/hub/spaces-sdks-docker
[16] Pallets Projects, "Flask 3.0 Documentation," 2024. [Online]. Available: https://flask.palletsprojects.com