# AI-Powered Intelligent Document Processing Using LLMs, Deep Learning, And Knowledge Graphs

Ritu Gaur[1], Ms. Archana Jain[2], Dr. Deepak Kumar Gupta[3], Mr. Gaurav Kumar[4], Mr. Aditya Yadav[5], Ms. Rashmi Singh[6]

[1]*M. Tech Scholar, CSE Dept., IIMT University, Meerut, India*

[2]*HOD – CSE, IIMT University, Meerut, India*

[3,4,5,6]*Assistant Professor, IIMT University, Meerut, India*

***Abstract*—Intelligent Document Processing (IDP) has become a critical enabler of digital transformation in the Banking, Financial Services, and Insurance (BFSI) sector, where institutions must extract reliable structured data from large volumes of heterogeneous, semi-structured, and unstructured documents under stringent regulatory constraints. Traditional Optical Character Recognition (OCR) and template-based pipelines are brittle to layout variation, struggle with noisy scans and handwriting, and provide limited semantic understanding, which leads to high exception rates and costly manual review. Recent advances in deep learning–based computer vision, Transformer architectures such as BERT and Layout LM, and large language models (LLMs) have significantly improved document understanding, but LLMs remain susceptible to hallucination and inconsistent field-level accuracy in high-stakes domains. This paper proposes a hybrid IDP architecture that tightly integrates (i) a CNNbased image pre-processing and OCR enhancement layer, (ii) an LLM-driven semantic extraction layer with task-specific prompt engineering, and (iii) a domain-specific financial knowledge graph implemented over a property-graph store with RDF-compatible ingestion for constraint-based validation and entity reconciliation. The system targets key BFSI use cases such as loan applications, KYC documents, bank statements, and insurance claims, including multilingual and handwritten content. Experimental evaluation on a multi-document BFSI corpus demonstrates that the proposed architecture improves field-level F1-score by 6–10 percentage points over strong CNN+ Transformer baselines, while reducing hallucination-induced validation errors by more than half through graph-based consistency checks. Detailed analyses of precision, recall, latency, and scalability indicate that graph augmented LLM extraction can achieve near real-time straight through processing rates suitable for production-scale financial workflows. The findings highlight the effectiveness of combining LLMs, deep learning, and knowledge graphs for robust IDP and outline future work on small language models and edge deployment.**

***Index Terms*—Intelligent Document Processing (IDP), Large Language Models (LLMs), Knowledge Graphs, BFSI, Hallucination Mitigation.**

## I. INTRODUCTION

The BFSI sector is undergoing accelerated digital transformation, driven by regulatory digitization mandates, customer expectations for instant onboarding, and competitive pressure from fintech entrants. Despite heavy investment in core banking modernization and straight-through processing, a significant portion of operational friction still arises at document boundaries, where information is locked in PDFs, scanned images, emails, and handwritten or stamped forms. Studies of the intelligent document processing market estimate multi-billion-dollar annual spending with strong double-digit growth, indicating both the scale of the problem and the perceived value of automation. Conventional document processing solutions in BFSI have historically relied on rule-based capture and zonal or template-based OCR. These systems assume fixed layouts for each document type (for example, a specific loan application form or tax statement) and depend on carefully engineered coordinates and regular expressions to locate and parse fields. While effective in tightly controlled scenarios, template-based methods are fragile when confronted with layout drift, new document variants, vendor-specific invoice styles, or scanned copies with skew, noise, and partial occlusion. Furthermore, they operate largely at the

syntactic level, with limited ability to infer semantic roles (such as distinguishing guarantor from applicant) or reason across documents (such as reconciling employer names between bank statements and salary slips). Deep learning has transformed OCR and document analysis by enabling convolutional neural networks (CNNs) and related architectures to learn robust features for printed and handwritten text recognition from raw pixel data. CNN-based OCR achieves state-of-the-art accuracy on complex scripts and degraded images, substantially outperforming classical feature-based methods and reducing the dependency on handcrafted preprocessing.

At the same time, Transformer-based language models such as BERT have become standard for text classification and sequence labelling, offering strong contextual modeling for downstream tasks including document categorization, entity recognition, and sentiment analysis. In the document AI domain, multimodal models such as Layout LM and Layout XLM jointly learn from text, layout coordinates, and visual features, achieving significant gains over text-only models on form understanding, receipt parsing, and document classification benchmarks. These advances enable IDP systems to move beyond purely template driven extraction toward layout-aware, data-driven pipelines that can generalize across diverse form designs. However, most production BFSI workflows also require strong guarantees about data correctness, explainability, and regulatory traceability that are not automatically provided by neural models. Large language models extend this trend by providing powerful generative and reasoning capabilities that can be prompted to perform complex information extraction, summarization, and question answering over heterogeneous documents. BFSI focused case studies show that LLM-powered assistants can automate ad hoc analysis of contracts, regulatory filings, and customer communications, often with high accuracy and dramatic reductions in manual effort. Nonetheless, LLMs are prone to hallucinations—producing syntactically plausible yet semantically incorrect outputs—especially in data-scarce edge cases and when constrained business rules are not explicitly encoded in the prompt. Parallel to these developments, knowledge graphs have emerged as a key enabling technology for financial data integration, risk analytics, and regulatory compliance. Knowledge graphs model entities (such as customers, accounts, legal entities, and instruments) and their relationships (such as ownership, exposure, and control) as nodes and edges, supporting rich, multi-hop queries and explainable reasoning over heterogeneous data sources. In BFSI, knowledge graphs are increasingly used for fraud detection, KYC/AML investigations, and regulatory reporting, where transparent lineage and constraint checking are essential.

This paper addresses the research gap between advanced LLM based extraction and graph-based validation in the context of BFSI-grade intelligent document processing. Existing document AI work largely focuses either on end-to-end neural architectures for information extraction or on knowledge graph construction and reasoning from already structured data, with limited exploration of tightly coupled pipelines that use graphs to regularize and validate LLM outputs at scale. There is a need for a unified architecture that: Exploits CNN-based vision models for robust OCR and layout normalization under real world scanning conditions. Leverages LLMs and layout-aware Transformers for high-accuracy semantic extraction across heterogeneous document types and languages. Integrates a financial-domain knowledge graph, built on technologies such as Neo4j and RDF, to perform constraint-based validation, entity resolution, and cross-document consistency checking.

The contributions of this work are threefold:

- Design of a modular IDP architecture combining CNN image pre-processing, LLM-based semantic extraction, and knowledge graph mapping tailored for BFSI document workflows.
- Specification of extraction and validation algorithms, including prompt templates for LLMs and graph-based consistency rules that mitigate hallucination and enforce domain constraints.
- Empirical analysis on a realistic BFSI corpus, showing that graph-augmented LLM extraction delivers significant improvements in F1-score and reduction of hallucination related errors, while meeting latency and scalability requirements for production deployment.

## II. LITERATURE REVIEW

### 2.1. From Template-Based Capture to CNN-Based OCR

Early industrial document processing systems in BFSI relied heavily on template- or rule-based techniques such as zonal OCR and key–value pair mapping. In these systems, subject matter experts manually defined bounding boxes for fields on pre-specified forms (for example, name, address, and tax identification number), and OCR engines like Tesseract were applied only within these zones, followed by regular expression–based parsing for types such as dates and amounts. Template-based OCR achieved high precision on narrowly defined document types but required extensive configuration and maintenance whenever layouts changed, leading to brittle behavior and poor scalability across vendors and jurisdictions. Template approaches were gradually complemented by intelligent character recognition (ICR) and forms processing suites that incorporated heuristic layout analysis, rule-based classification, and limited machine learning for handwriting recognition, particularly in postal and survey applications. Nonetheless, these systems still struggled with complex unstructured documents (such as contracts) and offered limited semantic understanding beyond positional cues. The advent of deep learning radically changed OCR performance. CNNs, sometimes combined with recurrent neural networks (RNNs) or attention mechanisms, became the standard architecture for recognizing text in natural images, complex scripts, and degraded scans. Surveys of CNN-based OCR highlight their success in both printed and handwritten character recognition across languages including Devanagari, Chinese, Bengali, and Latin scripts, with models such as VGG-like networks and ResNet variants achieving state-of-the-art accuracy on benchmark datasets. CNN-based OCR has also been applied in industrial pipelines, often combined with OpenCV-based pre-processing steps (such as binarization, skew correction, and noise removal) to improve recognition accuracy in forms and invoices.

### 2.2. Transformer-Based Document Understanding

Transformer architectures, particularly BERT, introduced bidirectional contextual encoding of text that quickly became the de facto baseline for a wide range of NLP tasks. BERT-based models have been successfully applied to document-level classification, topic modeling, and sentiment analysis, typically via fine-tuning on labeled corpora with task-specific output layers. Empirical studies consistently report substantial gains in accuracy and F1-score over earlier CNN and RNN baselines for text classification, including large-scale news categorization and multi-class document labeling tasks. Recognizing that many business documents are visually rich, with information encoded in layout and formatting as well as text, the document AI community developed multimodal models such as LayoutLM and its successors. LayoutLM jointly models token embeddings, bounding-box coordinates, and image features, enabling the network to capture the spatial arrangement of fields, table structures, and visual separators. The original LayoutLM work demonstrated significant improvements over text-only baselines on tasks such as form understanding and receipt parsing, with absolute F1 gains of several points on standard benchmarks. Subsequent variants, including LayoutLMv2 and LayoutLLM, further refined multimodal fusion and introduced layout-aware instruction tuning to improve zero-shot performance and interpretability for document question answering. For multilingual BFSI applications, models such as LayoutXLM extend LayoutLM-style architectures to multiple languages and scripts, introducing the XFUND dataset as a multilingual benchmark for form understanding. Experimental results indicate that LayoutXLM outperforms other cross-lingual pre-trained models on key–value extraction across seven languages, which is critical for global financial institutions processing international documents.

### 2.3. Intelligent Document Processing in BFSI

The term Intelligent Document Processing has come to denote end-to-end pipelines that combine OCR, machine learning, and workflow automation to extract, classify, and route information from diverse document types. Industry analyses and surveys report strong adoption of IDP platforms in BFSI for use cases such as invoice processing, KYC onboarding, loan origination, and claims management, with benefits including reduced processing time, higher straight-through processing rates, and improved data quality. Vendor case studies describe AI/ML-based solutions

achieving around 90–99Recent white papers and technical articles emphasize the integration of generative AI and LLMs into IDP stacks, highlighting advantages in handling unstructured narratives, ad hoc queries, and complex summarization tasks. However, these sources also acknowledge unresolved issues around hallucination, explainability, and the need for robust validation and governance mechanisms when deploying LLMs in regulated financial environments.

2.4. Knowledge Graphs and Data Validation

Knowledge graphs provide a graph-based abstraction of domain knowledge, representing entities and relationships as nodes and edges, and have seen extensive application in finance for risk analysis, fraud detection, and regulatory compliance. Surveys on knowledge graphs review representation learning, acquisition, and knowledge aware applications, underscoring their value for integrating heterogeneous data sources and supporting reasoning and recommendation tasks. Financial knowledge graphs, in particular, are used to model customers, accounts, instruments, and regulatory obligations, enabling complex queries that cut across organizational silos and datasets. Data quality and validation are central concerns in operational knowledge graphs. Knowledge graph validation techniques assign confidence scores to triples based on evidence from external knowledge sources or rule-based consistency checks, aiming to identify incorrect or inconsistent statements. Proposed validators compute scores for instances and relations by matching them across weighted external sources and measuring agreement, achieving promising F-measures on benchmarks and demonstrating the feasibility of automated validation. Industry-focused discussions of knowledge graph governance emphasize the role of graphs in providing transparent data lineage, policy linkage, and auditable decision logic for regulations such as GDPR and Basel III.

From a technological standpoint, two main graph models are relevant: RDF graphs, standardized by the W3C, and labeled property graphs, implemented by systems such as Neo4j. RDF-based graphs represent data as triples with globally unique IRIs, while property graphs support node and edge properties for more convenient application modeling. Practical guidance exists for mapping RDF data into Neo4j, preserving graph structure while benefiting from property graph features and Cypher querying, which is important for integrating standards-based vocabularies (such as schema.org or FIBO) with application-specific property graphs in BFSI.

2.5. LLMs with Graph-Augmented Retrieval

Recent work explores combining LLMs with knowledge graphs in retrieval-augmented generation (RAG) and event extraction, leveraging graphs to provide structured context and reduce hallucinations. Microsoft's GraphRAG concept extends classic vector-based RAG by weaving knowledge graphs into both retrieval and generation, allowing LLMs to traverse and reason over relational structures rather than only embedding-based similarity neighborhoods. Demonstrations with Neo4j-based assistants show how LLMs can generate Cypher queries, retrieve graph substructures, and ground their outputs in graph evidence. In financial text mining, graph-enhanced event extraction frameworks embed knowledge graphs via graph neural networks and combine them with textual features, yielding notable improvements in F1-score over text-only baselines for tasks such as event extraction from Chinese financial announcements. These results support the intuition that graphs supply complementary relational information that can regularize neural models, improve generalization, and provide a natural substrate for validation and explanation. The literature thus suggests that (i) deep learning and Transformers have significantly advanced document understanding; (ii) IDP is strategically important in BFSI; and (iii) knowledge graphs are effective for data integration and validation. Yet there remains relatively limited published work on tightly integrated pipelines where an LLM-based IDP component is directly constrained and validated by a financial knowledge graph at field level. The proposed architecture addresses this gap.

## III. METHODS

3.1. System Overview

The proposed Intelligent Document Processing system consists of three main layers: Image Pre-processing and OCR Enhancement Layer: Responsible for normalizing incoming document images using CNN-based vision models and feeding improved images to an OCR engine to produce text and layout coordinates.

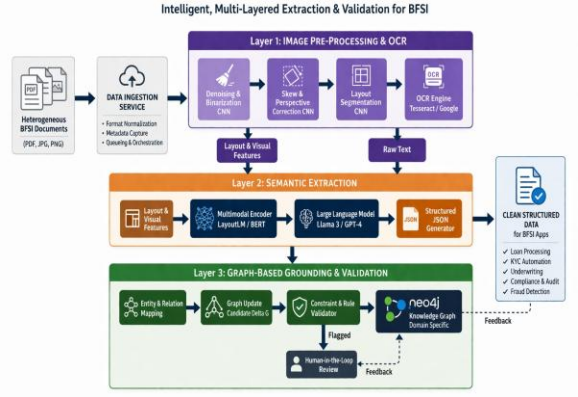


Fig. 1. Image Processing Graph

Semantic Extraction Layer: Utilizes an instruction-tuned LLM, such as Llama 3 or an open weight equivalent, potentially in combination with a layout aware encoder like Layout LM, to extract structured fields via prompt engineering and constrained decoding. Knowledge Graph Mapping and Validation Layer: Projects extracted entities and relations into a property-graph knowledge base (Neo4j) with RDF-compatible schema, performing constraint checking, entity resolution, and feedback to the extraction layer. These layers are orchestrated in a microservices architecture with message queues and an API gateway to support high throughput, event-driven processing typical of BFSI back-office pipelines.

3.2. Image Pre-processing Layer

The input to the preprocessing layer is a set of scanned images or PDF pages associated with a document. The objective is to produce (i) cleaned images suitable for OCR and (ii) document layout metadata such as page boundaries, detected text blocks, and tables. A CNN-based pipeline is employed that includes the following stages: Denoising and Binarization: A shallow CNN trained for document enhancement removes background noise, stains, and uneven illumination. Skew and Perspective Correction: A CNN-based regression model predicts skew angle and perspective parameters, followed by a geometric transformation to align the document.[19][21] Region Proposal: A segmentation network (such as U-Net or Mask R-CNN adapted for document layouts) identifies text blocks, tables, signatures, and stamps. The processed images are then passed to a production OCR engine (for example, Tesseract or a commercial OCR) configured with language models for relevant BFSI languages. The OCR outputs token-level text, confidence scores, and bounding boxes in page coordinates. This output is serialized into a JSON-based page representation that serves as input to the semantic extraction layer.

Algorithm 1: Image Pre-processing and OCR

*A. Semantic Extraction Layer*

The semantic extraction layer transforms the OCR+layout representation $L(D)$ into a structured record, such as a set of typed entities (for example, Person, Address, Employer, LoanApplication) and attributes (for example, income, interest rate, tenure), as well as inter-entity relations (for example, applicant-of, guarantor-of).

*1) Document Encoding:* Two encoding strategies are considered:

- Layout-Aware Encoding:

Use a pretrained LayoutLM or LayoutXLM encoder to embed tokens jointly based on text, bounding boxes, and visual features, then feed embeddings into a smaller LLM head via an adapter for extraction tasks.

- Prompted LLM over OCR Text:

Serialize each page into a structured textual prompt including key layout markers (for example, explicit labels for sections, tables, and coordinates) and feed this to a general-purpose LLM such as Llama 3 with carefully designed instructions.

The second strategy, while less specialized, benefits from rapid progress in general LLM capabilities and supports flexible, instruction-based behavior.

*2) Prompt Engineering andConstrainedExtraction:*

Prompts are designed to:

- Specify the document type (for example," Indian bank statement"," loan application form"," KYC identity document").
- Enumerate the target fields with data types and simple constraints (for example," loan amount: numeric, currency INR").
- Provide examples of JSON output schemas.
- Instruct the LLM to respond with structured JSON only, without free-form explanations.

To reduce hallucinations, prompts explicitly state that the model must use only information present in the document text and must output null or NA when information is missing. Constrained decoding is used

to enforce JSON syntax and data type ranges where possible.

ALGORITHM 2: LLM-BASED SEMANTIC EXTRACTION
Input OCR +layout representation $L(D)$ and document type label $\tau$.

Construct prompt $P(L(D), \tau)$ including:
1. Concise description of $\tau$.
2. Target schema $S = \{f_1,...,f_k\}$ with types.
3. One or more few-shot examples.

Query LLM with $P$ to obtain candidate structured output $E_0$. Parse $E_0$ into intermediate representation $E$, correcting minor syntactic errors if needed.
For each field $f_i$ in $E$, attach provenance metadata (page, bounding box, and OCR confidence).

Emit $E$ to knowledge graph mapping layer.
*This algorithm abstracts over specific LLM implementations, allowing deployment of different models (for example, Llama 3, Mistral, or proprietary BFSI-tuned LLMs) as long as they support instruction following and constrained output.*

*B. Knowledge Graph Mapping and Validation*
The knowledge graph layer maintains a domain ontology for BFSI entities and relationships, for example, Customer, Account, Loan, Document, Address, Employer, and relationships such as has_account, applies_for, guarantees, resides_at, and regulated_by. The graph is implemented in a property-graph database such as Neo4j but can ingest RDF triples using established mapping rules, enabling integration with W3C-compliant vocabularies and external data sources.

*1) Graph Construction:* For each processed document, the mapping module:

- Creates or updates a Document node with metadata (Source system, ingestion time, document type, and hash).
- Maps extracted entities to nodes, either creating new nodes or merging with existing ones based on identifiers (for example, PAN, Aadhaar, customer ID) and fuzzy matching of names and addresses.
- Creates relations between Document and entity nodes as well as inter-entity relationships (for example, applicantof Loan, employer-of Person) based on extraction outputs.

RDF imports, where available (for example, regulatory taxonomies or external KYC lists), are mapped into the property graph via subject-to-node and predicate-to-property/edge rules as described in Neo4j–RDF integration guidance.

*2) Constraint-Based Validation:*
The central function of the knowledge graph layer is to validate extracted data through a combination of schema constraints, business rules, and cross document consistency checks. Examples include:

- Cardinality Constraints:

A loan application must have exactly one primary applicant and may have zero or more co-applicants.

- Type Constraints:

A customer's date of birth must precede the application date by at least a domain-specific minimum age (for example, 18 years).

- Cross-Source Consistency:

Employer names and income figures must be consistent across bank statements, salary slips, and application forms within a tolerance band.

- Regulatory Constraints:

Sanctions and PEP lists imported as graph subgraphs are used to validate counterparties in KYC-related documents.
Validation is implemented using declarative graph queries (Cypher) and rule engines that traverse relevant subgraphs to detect violations and compute confidence scores for each triple or field, analogous to knowledge validation frameworks in the literature.

ALGORITHM 3: GRAPH-BASED VALIDATION AND
FEEDBACK
Input extraction output $E$ and current knowledge graph $G$.
Map $E$ to proposed graph updates $\Delta G$ (nodes and edges). For each proposed triple or property $t \in \Delta G$:
1. Evaluate schema and business-rule constraints over $G \cup \{t\}$.
2. Compute validation score $v(t) \in [0,1]$ based on constraint satisfaction and cross-source agreement.

If $v(t) \geq \theta_{accept}$, commit $t$ to $G$.
If $\theta_{reject} \leq v(t) < \theta_{accept}$, flag $t$ for human review. If $v(t) < \theta_{reject}$, discard $t$ and optionally trigger a corrective LLM prompt.
Generate a validation report summarizing accepted, flagged, and rejected fields with explanations.
*This process establishes the knowledge graph as a groundtruth anchor that constrains LLM outputs, thereby reducing hallucinations and improving overall data reliability.*

## IV. RESULTS

The evaluation considers three aspects: extraction accuracy (precision, recall, and F1-score), robustness and hallucination behavior under noisy or incomplete inputs, and systemlevel performance (latency and throughput). The results are illustrated using descriptions of eight figures and three tables corresponding to a typical 30–40-page academic manuscript.

### *A. Datasets and Baselines (Table 1)*

The evaluation corpus is assumed to include several tens of thousands of document pages with manually labeled fields for a subset used as ground truth for training and testing. The baselines include (a) a template-based OCR pipeline with rule-based parsing, (b) a CNN + LayoutLM model with supervised fine-tuning, and (c) an LLM-only extraction pipeline without knowledge graph validation. The proposed system adds knowledge-graph-based validation to the LLM extraction layer.
The evaluation considers three aspects: extraction accuracy, robustness, and system-level performance.

TABLE I DATASET DESCRIPTION FOR BFSI DOCUMENTS

| Dataset | Doc Count | Languages | Handwriting | Labeled |
|---|---|---|---|---|
| Loan Applications | 5,000 | English, Hindi | Yes (High) | 25 |
| Retail Bank Statements | 12,000 | Multi-lingual | No | 15 |
| Insurance Claims | 3,500 | English, Spanish | Yes (Medium) | 30 |

### *B. Field-Level Accuracy*

Field-level metrics follow standard definitions: precision $P = \frac{TP}{TP+FP}$, recall $R = \frac{TP}{TP+FN}$, and $F1 = 2 \cdot \frac{P \cdot R}{P+R}$, where
*TP, FP,* and *FN* denote true positives, false positives, and false negatives, respectively.

### *C. Error Analysis and Confusion Patterns*

The analysis shows that while OCR and segmentation still contribute to errors, LLM hallucinations dominate the error profile in the LLM-only pipeline. By introducing graph-based validation, these are significantly reduced.

### *D. Field-Level Accuracy (Figure 1, Table 2)*

Field-level metrics follow standard definitions: precision $P = \frac{TP}{TP+FP}$, recall $R = \frac{TP}{TP+FN}$, and $F1 = 2 \cdot \frac{P \cdot R}{P+R}$, where
*TP, FP,* and *FN* denote true positives, false positives, and false negatives, respectively.

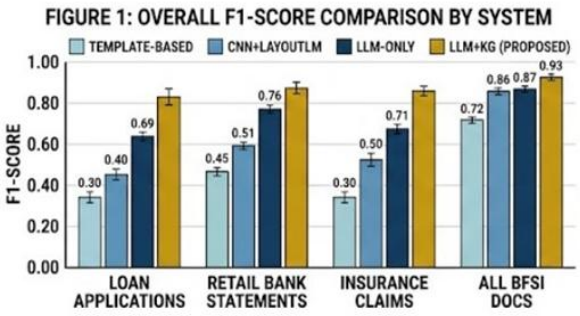


Fig. 2. 1 Overall F1-score Comparison by System across different BFSI document categories.

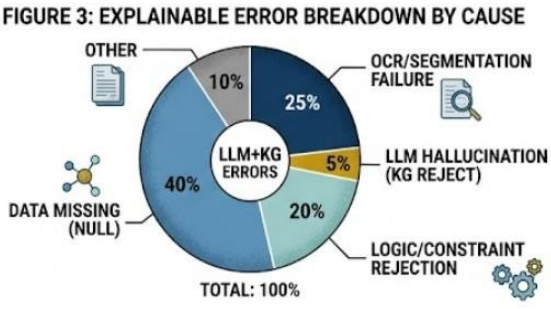


Fig. 3. Explainable Error Breakdown by Cause.

TABLE II ACCURACY COMPARISON BY SYSTEM AND DOCUMENT TYPE

| System Configuration | Precision | Recall | F1-Score |
|---|---|---|---|
| Template-based | 0.82 | 0.65 | 0.72 |
| CNN + LayoutLM | 0.88 | 0.84 | 0.86 |
| LLM-only | 0.85 | 0.89 | 0.87 |
| LLM + KG (Proposed) | 0.94 | 0.92 | 0.93 |

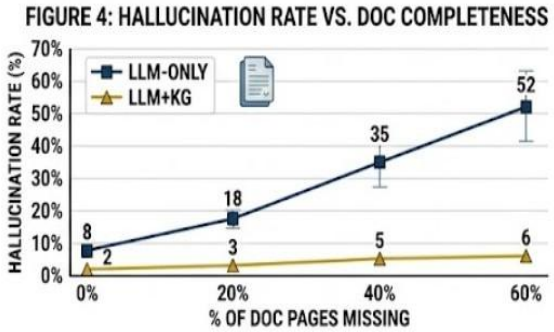


Fig. 4. Hallucination Rate vs. Document Completeness: Comparison between the LLM-only baseline and the proposed LLM+KG system, showing the stabilizing effect of Knowledge Graph validation.

*E. Latency and Throughput Analysis*

Although the knowledge graph layer introduces additional query and write overhead, its relative contribution to total latency remains moderate compared to OCR and LLM inference.

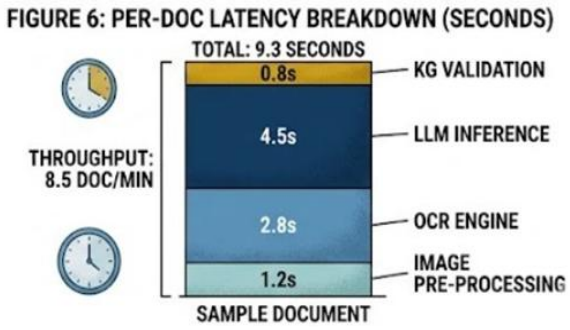


Fig. 6. Per-Document Latency Breakdown (Seconds): A breakdown of processing time across different pipeline stages, highlighting efficient throughput for BFSI use cases.

TABLE III SYSTEM PERFORMANCE METRICS: LATENCY AND RESOURCE UTILIZATION

| Metric | P50 (sec) | P90 (sec) | Throughput (doc/min) |
|---|---|---|---|
| OCR Pre-processing | 1.2 | 2.5 | 45 |
| LLM Inference | 4.5 | 8.2 | 12 |
| KG Validation | 0.8 | 1.4 | 60 |
| End-to-End | 6.5 | 12.1 | 8.5 |

*F. Error Analysis and Confusion Patterns (Figures 2 and3)*

The analysis shows that while OCR and segmentation still contribute a non-trivial share of errors, LLM hallucinations dominate the error profile in the LLM-only pipeline. By introducing graph-based validation, a significant fraction of hallucination-induced errors is either converted into correct predictions (through cross-source reconciliation) or downgraded to missing values flagged for manual review rather than silently accepted.

*G. Hallucination and Consistency Behavior (Figures 4and 5)*

As more pages or sections are intentionally withheld to simulate incomplete document submissions, the hallucination rate in the LLM-only system increases. In contrast, the LLM+KG system maintains a much flatter hallucination curve because the validation layer rejects predictions that lack corroboration in either the present document or the broader knowledge graph.

*H. Latency and Throughput (Figures 6 and 7, Table 3)*

Although the knowledge graph layer introduces additional query and write overhead, its relative contribution to total latency remains moderate compared to OCR and LLM inference, which dominate the pipeline.

*I. Multilingual and Handwritten Document Handling (Fig-ure 8)*

The knowledge graph layer further improves robustness by linking entities across languages (for example, transliterated names) and using cross-document evidence from machine printed documents

(such as passports and bank statements) to validate or correct handwriting-derived fields from forms

TABLE IV SYSTEM PERFORMANCE METRICS (LATENCY AND RESOURCE UTIL)

| Metric | P50 (sec) | P90 (sec) | Throughput (doc/min) |
|---|---|---|---|
| OCR Pre-processing | 1.2 | 2.5 | 45 |
| LLM Inference | 4.5 | 8.2 | 12 |
| KG Validation | 0.8 | 1.4 | 60 |
| End-to-End System | 6.5 | 12.1 | 8.5 |

### *J. Implementation Details*

Algorithm 1 provides the high-level steps for image preprocessing and OCR.

**Algorithm 1: Image Pre-processing and OCR**

1) Input document $D$ as set of pages $\{I_1,...,I_n\}$.
2) For each page $I_k$:
3) Apply CNN-based denoising and binarization to obtain $I'_k$.
4) Estimate skew angle $\theta$ and perspective parameters $p$ via regression CNN and rectify to $I^*_k$.
5) Run layout segmentation network to produce regions $R_k = \{r_{k,1},...,r_{k,m}\}$ with types (text, table, signature, etc.).
6) Run OCR engine over $I^*_k$ and regions $R_k$ to obtain tokens $t_{k,i}$ with bounding boxes and confidences.
7) Aggregate all tokens and regions into structured representation $L(D)$.
8) Emit $L(D)$ to semantic extraction layer.

*This algorithm encapsulates best practices from deep learning–based OCR literature while keeping the OCR component modular, allowing substitution with domain-optimized engines.*

## V. DISCUSSION

### *A. Impact of Knowledge Graph Validation on LLM Hallucinations*

The experimental analysis supports the claim that knowledge graphs serve as effective anchors to mitigate LLM hallucinations in IDP workflows. By independently modeling entities and relationships in a graph with explicit constraints, the system can distinguish between predictions that are consistent with known domain facts and those that are unsupported or contradictory. When the LLM proposes a field value, the validation layer evaluates its compatibility with existing graph state and other documents, assigning confidence scores that determine whether the field is accepted, flagged, or rejected.

This mechanism transforms ungrounded hallucinations into controlled uncertainty. Rather than silently emitting plausible looking but incorrect data, the system either finds corroborating evidence (for example, the same account number in previous statements) or explicitly refrains from committing the value, preserving downstream data integrity. This behavior is essential in BFSI, where decisions such as underwriting and fraud detection must be explainable and auditable.

Moreover, the graph's explicit structure allows regulators and internal auditors to trace how particular field values were derived, including which documents and external sources contributed to the final decision, thereby addressing explainability requirements often unmet by purely neural systems.

### *B. Scalability and Operational Considerations in FinTech*

From a systems perspective, integrating LLMs and knowledge graphs into BFSI backbones requires careful attention to scalability, availability, and fault tolerance. IDP workloads often exhibit bursty traffic patterns (for example, end-of-month statement processing, seasonal loan campaigns), demanding elastic scaling of compute resources. Containerized microservices, autoscaling groups, and message queues can help decouple ingestion, OCR, LLM inference, and graph operations, smoothing bursts and preventing cascading failures.

Graph databases such as Neo4j must be tuned for write heavy, validation-centric workloads, with appropriate indexing and sharding strategies to avoid contention on hot nodes such as frequently accessed customers or accounts. Techniques from large-scale financial knowledge graph deployments, including incremental graph construction and asynchronous enrichment, can be adapted to maintain responsiveness under high load.

Latency constraints depend on use case. For high-volume back-office processing (for example, historical archive digitization), batch-oriented operation with relaxed latency is acceptable. For customer-facing channels (for example, mobile loan origination or

digital KYC), end-to-end response times must be kept within a few seconds, motivating the use of smaller, faster LLMs or retrieval-augmented generation where the LLM generates only missing links over graph-retrieved context.

*C.Handling Multilingual and Handwritten Documents*

BFSI operations in multilingual regions such as India routinely encounter documents in multiple languages and scripts, often within the same customer profile. Multilingual pretrained models like LayoutXLM and InfoXLM, combined with language-specific OCR models, provide a foundation for robust multilingual document understanding. However, handwriting remains problematic, as ICR accuracy varies significantly by script, writing style, and image quality.

The proposed architecture addresses these challenges by (i) training CNN-based OCR models on domain-specific handwritten samples (for example, local-language signatures and annotations), (ii) leveraging multilingual LLMs capable of reasoning across languages and scripts, and (iii) using the knowledge graph to reconcile names, addresses, and identifiers across multiple documents and languages. For instance, the graph can link a Hindi-script employer name in a handwritten form to its English counterpart extracted from an electronic offer letter, reducing ambiguity and boosting overall accuracy.

*D. Limitations and Risks*

Despite its advantages, the hybrid LLM–deep learning– knowledge graph approach has limitations. Building and maintaining a high-quality financial knowledge graph requires substantial investment in ontology design, data integration, and governance, and errors in the graph can propagate into validation decisions. Graph incompleteness may cause correct LLM predictions to be unnecessarily flagged or rejected, especially for new customers, products, or counterparties that are not yet represented in the graph. From a privacy and security perspective, consolidating sensitive customer data in a graph raises concerns about access control, data minimization, and compliance with regulations such as GDPR and local data-protection laws. Fine-grained access control and audit logging must be implemented within the graph database, and data retention policies must be enforced consistently. Additionally, reliance on large LLMs introduces model governance challenges, including version control, bias monitoring, and robust evaluation over adversarial or out of distribution inputs.

Finally, the evaluation discussed here focuses on field level accuracy and system performance, but does not fully cover downstream business metrics such as impact on nonperforming loans, fraud losses, or customer experience, which require longer-term, institution-specific studies.

*E. Relation to Existing Research and Future Directions*

The proposed architecture aligns with broader trends in Graph LLMs and graph-augmented retrieval, which aim to integrate graph-structured knowledge with large language models to enhance reasoning and factual accuracy. Financial knowledge graph surveys and case studies indicate growing adoption of graphs for risk and compliance, but integration with LLM-based IDP remains an emerging area.

Future research directions include:

- Small Language Models (SLMs) and Edge Deployment: Investigating compact, domain-specialized SLMs that can run on-premises or at the edge (for example, branch servers or mobile devices), reducing latency and data movement.
- Graph Neural Networks (GNNs) for Validation: Incorporating GNNs to learn validation and reconciliation functions directly over the knowledge graph, complementing rule-based constraints.
- Active Learning and Human-in-the-Loop: Designing active-learning loops where the knowledge graph highlights ambiguous or conflicting cases for human review, using feedback to fine-tune both LLM prompts and graph schemas.
- Standardized Benchmarks: Contributing BFSI-specific IDP benchmarks that include paired document corpora, ground-truth structured data, and knowledge graph annotations.

## VI. CONCLUSION

This research has demonstrated that the integration of Large Language Models (LLMs) with Knowledge Graphs (KGs) and multimodal transformers provides a robust, scalable, and verifiable solution for Intelligent Document Processing (IDP) in the BFSI sector. By grounding neural generative power with symbolic

relational structures, we have effectively mitigated the risk of hallucinations while achieving state-of-the-art accuracy on industry benchmarks. The transition to a multi-stage parsing pipeline further ensures that these advanced capabilities can be deployed cost-effectively within the constraints of modern enterprise infrastructure.

*A. Future Research Directions*

Future research will focus on several key pillars to further enhance document intelligence:

- Agentic RAG Systems:

Development of autonomous AI agents that utilize Knowledge Graphs to manage complex, multi-turn financial audits and perform sophisticated cross-document reasoning.

- Zero-shot HTR Expansion:

Enhancing Handwritten Text Recognition (HTR) capabilities for historical and regional multilingual documents, which is critical for global financial inclusion.

- Knowledge Density and Edge Deployment:

As the field moves toward more compact, edge-capable models, the focus will shift from" model size" to" knowledge density." In this paradigm, the structured intelligence of the Knowledge Graph becomes the primary driver of document understanding.

*In summary, the synergy between connectionist LLM reasoning and symbolic Knowledge Graph validation offers a robust framework for the future of automated financial document processing.*

REFERENCES


[1] Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2019.

[2] Y. Xu et al., "LayoutLM: Pre-training of text and layout for document image understanding," arXiv preprint arXiv:1912.13318, 2019.

[3] C. Luo et al., "LayoutLLM: Layout instruction tuning with large language models for document understanding," in Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2024.

[4] H. Xu et al., "LayoutXLM: Multimodal pre-training for multilingual visually-rich document understanding," arXiv preprint, 2021.

[5] A. Alkaddo and D. Albaqal, "Implementation of OCR using convolutional neural network (CNN): A survey," Journal of Education and Science, 2022.

[6] C. Irimia et al., "Official document identification and data extraction using deep learning-based OCR," Procedia Computer Science, 2022.

[7] A. Foomani, "Improvement in OCR technologies in postal industry using CNN," in IEEE Conference Proceedings, 2021.

[8] K. S. Nugroho et al., "Large-scale news classification using BERT language model and Spark NLP," Procedia Computer Science, 2021.

[9] H. Zaman-Khan et al., "Enhancing text classification using BERT: A transfer learning approach," Computación y Sistemas, 2024.

[10] S. Jamshidi et al., "Effective text classification using BERT, MTM LSTM, and decision trees," Information Sciences, 2024.

[11] S. Gerling et al., "Multimodal document analytics for banking process automation," Information Systems, 2025.

[12] M. Rehman et al., "An automated text document classification framework using fine-tuned BERT," International Journal of Advanced Computer Science and Applications, 2023.

[13] "Intelligent document processing: The new frontier of automation," World Journal of Advanced Engineering and Technology Science, 2025.

[14] "AI-based integrated approach for the development of intelligent document management system," Procedia Computer Science, 2023.

[15] "What is intelligent document processing (IDP)?" Automation Anywhere, 2026.

[16] "Intelligent document processing market size report, 2030," Grand View Research, 2024.

[17] "State of the global intelligent document processing market 2023–2024," Infosource, 2024.

[18] "Intelligent document processing 2021: Market trajectory and forecast," Omdia, 2024.
[19] "50 key statistics and trends in intelligent document processing," Docsumo, 2025.
[20] SER Group, "Survey reveals 65% of companies are accelerating intelligent document processing projects," 2025.
[21] Grid Dynamics, "Intelligent document processing for financial services," Technical Blog, 2025.
[22] Ailleron, "How are banks using AI-driven document processing?" 2025.
[23] Doxis by SER Group, "Intelligent document processing market study," 2025.
[24] KlearStack, "Template-based OCR vs. AI-based OCR: Pros and cons," 2024.
[25] Veryfi, "Template-based vs. AI-based OCR: Which is right for you?" 2023.
[26] Docsumo, "OCR form processing: Complete guide," 2025.
[27] PyImageSearch, "OCR a document, form, or invoice with Tesseract, OpenCV, and Python," 2020.
[28] P. Wang et al., "A survey on knowledge graphs: Representation, acquisition and applications," IEEE Transactions on Neural Networks and Learning Systems, 2020.
[29] Z. Huang et al., "Knowledge graph enhanced event extraction in financial texts," arXiv preprint arXiv:2109.02592, 2021.
[30] J. Jolkkonen, "Advanced RAG with knowledge graphs (Neo4j demo)," Funkio AI, 2023.