Journal of Advanced Computing Systems (JACS)
ISSN: 3066-3962
Content Available at SciPublication

Improving Classification Accuracy for Unstructured Medical Documents via Multi-Engine OCR and Deep Learning Collaboration

Qiaomu Zhang
Computer Science, Rice University, TX, USA

DOI: 10.69987/JACS.2026.60201

Keywords

Medical document classification, Multi-engine OCR, Ensemble deep learning, Healthcare information extraction

Abstract

The exponential growth of unstructured medical documents poses significant challenges for healthcare information management. This study presents a novel multi-engine collaborative framework integrating diverse optical character recognition (OCR) technologies with ensemble deep learning classifiers to enhance document classification accuracy. The proposed approach adaptively selects optimal OCR engines based on document characteristics, extracts multi-source textual features, and employs confidence-weighted ensemble strategies. An experimental evaluation on a healthcare document dataset achieves 94.7% classification accuracy across clinical notes, diagnostic reports, laboratory results, insurance claims, and prescription forms, outperforming the strongest single-engine baseline (Engine-H) by 11.6 percentage points. The framework maintains an average processing time of 2.4 seconds per document while reducing computational consumption compared with parallel multi-engine execution. These findings validate the effectiveness of multi-engine collaboration for heterogeneous medical documentation systems.

1. Introduction

1.1. Background and Motivation

1.1.1. The growing volume of unstructured medical documents in healthcare

Healthcare institutions manage unprecedented volumes of unstructured documentation, including physician notes, laboratory reports, radiology interpretations, and administrative correspondence. Large-scale language models demonstrate promising capabilities for handling clinical narratives [1]. The United States healthcare system generates approximately 2.5 exabytes annually, with unstructured text comprising 80% of this information. This landscape encompasses handwritten prescriptions, scanned forms, typed summaries, and mixed-format reports.

Digital transformation creates opportunities for automated extraction and clinical decision support. Medical institutions require robust classification mechanisms to route documents through workflows, enable rapid retrieval, and support population analytics. Medical terminology complexity, domain abbreviations, and varied layouts present substantial challenges.

1.1.2. Challenges in automated document classification for clinical workflows

Automated classification encounters multiple obstacles. Handwritten notes exhibit penmanship variability with degradation under time pressure. Diagnostic imaging reports combine structured findings with narrative impressions[2]. Insurance documents incorporate tables, checkboxes, and free-text across standardized templates. Scanned quality varies based on paper condition, equipment, and age.

Single-engine OCR approaches struggle with this diversity. Standard systems optimized for print perform suboptimally on handwriting[3]. Specialized engines excel at cursive but underperform on tables. Applying multiple OCR engines creates processing bottlenecks in high-throughput environments.

1.2. Problem Statement

1.2.1. Limitations of single-engine OCR approaches for diverse medical documents

Single-engine deployments face fundamental limitations in processing heterogeneous documentation. Conventional technologies demonstrate accuracy degradation when encountering variations outside training distributions. Handwriting-specialized engines achieve 95% accuracy on cursive prescriptions but only 78% on printed laboratory reports[4]. This inconsistency undermines classification reliability across categories.

Static approaches prevent adaptation to document-specific characteristics. Medical documents exhibit diverse structural properties: radiology reports follow templated formats while progress notes contain free-form narratives. Applying uniform OCR processing results in suboptimal text extraction quality.

1.2.2. Accuracy and efficiency trade-offs in existing classification methods

Current systems face tension between accuracy requirements and operational efficiency constraints. High-accuracy deep learning classifiers demand substantial computational resources, creating processing latencies incompatible with real-time workflows. Healthcare institutions processing thousands of daily documents cannot afford multi-minute classification times.

Existing approaches prioritize either maximizing accuracy through ensemble methods or optimizing speed through simplified architectures. Accuracy-focused systems employing multiple parallel OCR engines consume excessive resources[5], while speed-optimized single-engine implementations tolerate unacceptable error rates. The medical domain requires accuracy above 95% to prevent clinical errors, yet latencies must remain below 5 seconds. Reconciling these demands necessitates intelligent coordination.

1.3. Research Objectives and Contributions

1.3.1. Proposed multi-engine collaborative framework

This research introduces a multi-engine collaborative framework dynamically coordinating OCR technologies with ensemble deep learning classifiers. The system employs adaptive engine selection based on rapid document assessment, directing each document to the appropriate OCR configuration. Quality-aware feature extraction integrates textual outputs from multiple engines, weighting contributions by their confidence scores [6].

The framework incorporates three innovations. A lightweight document profiler analyzes visual characteristics, predicting optimal OCR combinations. Multi-source feature fusion combines outputs from complementary engines, leveraging respective strengths while mitigating weaknesses. Confidence-calibrated ensemble classification adaptively weights predictions according to document-specific uncertainty estimates.

1.3.2. Key contributions and paper organization

This study makes four primary contributions. We develop adaptive multi-engine OCR coordination, selecting optimal pipelines based on document characteristics. We design layout-aware feature extraction preserving structural information during visual-to-textual transformation. We propose confidence-weighted ensemble learning dynamically adjusting classifier combinations. We provide a comprehensive empirical analysis comparing OCR-classifier configurations across diverse categories.

Section 2 reviews related work in medical document classification, OCR technologies, and ensemble strategies. Section 3 details our multi-engine framework. Section 4 presents experimental results and analysis. Section 5 concludes with findings and implications.

2. Related Work

2.1. Medical Document Classification Approaches

2.1.1. Traditional machine learning methods for clinical text

Early research employed conventional machine learning, including support vector machines, random forests, and logistic regression. These relied on manually engineered features such as TF-IDF representations, n-gram patterns, and medical concept ontology mappings[7]. Feature engineering required extensive domain expertise in identifying diagnostically relevant textual patterns. Bag-of-words assumptions discard crucial sequential information in clinical narratives.

Traditional approaches achieved moderate success on well-structured documents. Classification accuracy reached 85-88% for standardized radiology reports were templated language reduced vocabulary variability. Performance degraded when processing free-text physician notes that contained colloquialisms and abbreviations.

2.1.2. Deep learning advances in healthcare NLP

Neural network architectures revolutionized medical text processing by automating feature learning. Recurrent neural networks and long short-term memory networks captured sequential dependencies in clinical narratives[8]. Convolutional networks identified local textual patterns indicative of specific categories. These approaches eliminated manual feature engineering while achieving superior performance.

Transformer architecture and attention mechanisms enabled modeling of long-range dependencies in extended documents. Pre-trained language models adapted general linguistic knowledge to the medical domain through continued training on clinical corpora.

2.1.3. Pre-trained language models for medical documents

Large-scale pre-trained models achieved state-of-the-art performance across medical NLP tasks. Models fine-tuned on clinical text demonstrated strong transfer learning, adapting general language understanding to medical terminology and documentation patterns[9]. Domain-specific pre-training on millions of clinical notes enabled learning medical concept relationships, abbreviation expansions, and contextual usage patterns.

Pre-trained models exhibited varying performance across documentation types. Some architectures excelled at structured reports while others handled free-form narratives effectively. Computational requirements created deployment challenges in resource-constrained settings.

2.2. OCR Technologies for Healthcare Documents

2.2.1. Conventional OCR engines and their limitations

Commercial OCR engines employing traditional computer vision techniques dominated early digitization efforts. These systems used template matching, connected component analysis, and character pattern recognition to convert scanned images into machine-readable text [10]. Performance reached acceptable levels for high-quality printed documents. Accuracy degraded substantially when processing documents with quality issues such as fading or physical damage.

Conventional technologies struggled with heterogeneous documentation characteristics. Standard engines trained on printed text failed to reliably recognize handwritten content. Specialized medical vocabulary, including pharmaceutical names and anatomical terminology, exceeded the lexical coverage of general-purpose systems.

2.2.2. Handwriting recognition in medical prescriptions

Handwritten medical prescriptions present extreme challenges for automated extraction. Physicians' handwriting varies dramatically in legibility, with time pressure degrading script quality. Medical abbreviations compound recognition difficulty. Prescription forms combine handwritten medication names with printed patient information[11], requiring integrated recognition of mixed content types.

Deep learning approaches achieved substantial improvements over conventional methods. Convolutional recurrent neural network architectures demonstrated effectiveness in recognizing handwritten prescriptions through combined spatial feature extraction and sequential modeling. Despite advances, handwriting recognition accuracy remained below that of printed text.

2.2.3. Table detection and structured data extraction

Medical documents frequently incorporate tabular information presenting laboratory results, vital signs, or medication schedules. Extracting structured data requires detecting table boundaries, identifying row-column structures, and recognizing cell contents[12]. Traditional heuristic approaches relied on identifying horizontal and vertical line patterns but failed on borderless tables common in medical documentation.

Neural network approaches achieved superior performance by learning to recognize table structures. Models that combine visual feature extraction with layout analysis successfully identify table regions within complex document pages.

2.3. Ensemble and Multi-Engine Strategies

2.3.1. Ensemble learning for document classification

Ensemble learning combines predictions from multiple models, achieving superior performance compared to individual classifiers. Voting strategies aggregate outputs from diverse base classifiers, leveraging complementary strengths while mitigating weaknesses[13]. Bagging methods train multiple models on bootstrapped data samples, reducing prediction variance. Stacking ensembles learn meta-models optimally combining base classifier outputs.

Medical document classification benefited substantially from ensemble approaches. Different classifier architectures excelled at distinct categories: CNN-based models performed well on structured reports, while LSTM networks handled narrative notes effectively.

2.3.2. Multi-engine OCR fusion techniques

Multi-engine OCR combines outputs from diverse text extraction systems, improving overall recognition accuracy. Voting-based fusion selects the character recognition result that occurs most frequently across engines. Confidence-weighted fusion combines OCR outputs based on engine-specific certainty scores[14]. Learning-based fusion employs machine learning to predict optimal character selections from competing hypotheses.

Research on handwritten prescription recognition showed that ensemble methods could improve accuracy by combining outputs from different algorithms. Multi-engine approaches demonstrated particular effectiveness in processing degraded documents. Studies revealed that combining multiple analysis techniques enhanced robustness across document variations[15].

3. Methodology

3.1. Multi-Engine OCR Coordination Framework

3.1.1. Engine selection criteria for different document types

The coordination framework begins with rapid document characterization and the determination of optimal OCR processing strategies. Visual feature extraction analyzes pixel-level characteristics, including contrast distribution, edge density, and textural patterns. Statistical analysis of connected components reveals whether the content consists of printed text, handwritten annotations, or a mix of formats.

Document profiling employs a lightweight convolutional neural network trained on labeled medical documents spanning clinical notes, diagnostic reports, prescriptions, and insurance forms. The profiler extracts 128-dimensional visual embeddings capturing document appearance characteristics. During training, K-means clustering identifies routing-oriented document clusters with shared visual properties, used solely for OCR engine selection and not the same as the downstream classification labels. In inference, the profiler assigns incoming documents to clusters based on embedding proximity.

The framework maintains a performance lookup table mapping document categories to optimal OCR engine configurations. This table contains empirically measured accuracy and processing time metrics for each engine-category combination. Three primary OCR engines serve distinct purposes: a standard print-oriented engine (Engine-P) optimized for typed text, a handwriting-specialized engine (Engine-H) trained on cursive medical scripts, and a table-detection engine (Engine-T) designed for structured data extraction.

For ambiguous documents exhibiting mixed characteristics, the framework employs a dual-engine strategy. Both the top-ranked and the second-ranked engines process the document in parallel, with outputs combined via quality-aware fusion.

3.1.2. Adaptive routing based on document characteristics

Document routing proceeds through a decision tree incorporating multiple characteristic dimensions. Primary branching evaluates whether content appears predominantly handwritten or printed using connected-component shape analysis. Handwritten content exhibits irregular character boundaries and variable baseline alignment, while printed text demonstrates uniform spacing.

Secondary routing considers document structural complexity. Layout analysis identifies tabular regions using Hough transform detection of horizontal and vertical line patterns. Documents containing substantial tabular content route to Engine-T. Form-based documents with checkbox regions receive specialized processing.

The routing mechanism incorporates confidence thresholding to trigger multi-engine processing. Single-engine outputs below 0.75 confidence automatically invoke secondary engines. The confidence metric aggregates OCR engine-reported token-level confidence scores:

$$\text{Confidence} = \frac{1}{N} \times \sum_{i=1}^{N} P(c_i)$$

where N represents the total number of recognized tokens (e.g., words) and P(c i) denotes the OCR engine-reported confidence score for token i.

3.1.3. Text extraction quality assessment mechanism

Quality assessment quantifies the reliability of OCR output without ground-truth annotations. The framework employs multiple quality indicators computed directly from extraction results. Lexicon compliance measures proportion of extracted tokens appearing in a medical terminology dictionary compiled from SNOMED-CT, RxNorm, and LOINC ontologies:

$$L = \frac{\text{Tokens\_in\_lexicon}}{\text{Total\_tokens}}$$

Character-level confidence aggregation produces document-wide quality scores reflecting OCR certainty. The framework computes confidence statistics,

including mean, minimum, and standard deviation, across all characters.

Language model perplexity provides another quality indicator based on linguistic coherence. A domain-adapted medical language model computes conditional probabilities for extracted text sequences:

$$P = \exp\left(-\frac{1}{M} \times \sum_{i=1}^{M} \log P\left(w_i \mid w_1, \ldots, w_{i-1}\right)\right)$$

The quality assessment module aggregates multiple indicators into composite scores to guide downstream processing. Documents exceeding quality thresholds proceed directly to classification. Marginal-quality documents undergo targeted re-extraction.

3.2. Feature Extraction and Representation

3.2.1. Combining textual features from multiple OCR outputs

Multi-engine OCR produces multiple text-extraction hypotheses, requiring systematic integration. The framework employs character-level fusion for precise control. When multiple engines process the same document, character alignment algorithms match corresponding positions by minimizing edit distance.

For each aligned character position, the fusion module evaluates competing hypotheses. Character selection employs confidence-weighted voting:

$$c = \arg\max_{c} \sum_{e \in \text{Engines}} w_e \times I(OCR_e = c) \times \text{Conf}_e$$

where w e represents engine weight, I() is indicator function, and Conf_e is confidence score.

Fusion incorporates linguistic context through language model rescoring. A medical language model evaluates the likelihood of candidate character sequences, favoring those that produce coherent medical terminology. Document layout analysis integrates structural information with textual content, representing documents as hierarchical structures that maintain spatial relationships.

3.2.2. Layout-aware feature encoding using attention mechanisms

Spatial position encoding augments textual features with location information, preserving document structure. Each text token receives position embeddings indicating coordinates on the page:

$$PE(x, y, 2i) = \sin\left(\frac{x}{10000^{\frac{2i}{d}}}\right)$$

$$PE(x, y, 2i+1) = \cos\left(\frac{y}{10000^{\frac{2i}{d}}}\right)$$

where x and y are pixel coordinates, i is the embedding dimension index, and d is the total embedding dimensionality.

Attention mechanisms enable selective focus on salient regions of documents during feature extraction. Multi-head self-attention computes relationships between all token pairs, identifying dependencies between distant document sections. The framework employs hierarchical attention operating at multiple granularities: token-level, sentence-level, and section-level.

Visual features from original document images complement extracted text in multimodal representations. Convolutional encoders process document images, producing visual embeddings capturing layout, formatting, and graphical elements not represented in text.

3.3. Classification with Ensemble Deep Learning

3.3.1. Base classifier architecture design

The classification framework employs specialized base classifiers optimized for different document characteristics. A CNN-based classifier processes documents as sequences of token embeddings, applying convolutional filters to detect local textual patterns. Filters of varying kernel sizes capture patterns from specific terminology to multi-word phrases.

A bidirectional LSTM classifier handles sequential dependencies in narrative medical documents. The LSTM architecture processes tokens sequentially in both directions, building contextualized representations incorporating preceding and following context. The hidden state dimensionality of 256 provides sufficient capacity for capturing complex patterns.

A transformer-based classifier leverages pre-trained medical language representations fine-tuned for document classification. The base transformer employs 12 attention layers with 768-dimensional hidden representations. Fine-tuning adjusts pre-trained weights through supervised training on labeled documents.

Base classifiers receive category-specific training that emphasizes their respective strengths, creating complementary error patterns that enable effective ensemble combination.

3.3.2. Ensemble fusion strategies for accuracy improvement

Ensemble fusion aggregates predictions from multiple base classifiers through weighted averaging of class probabilities:

$$P_{ensemble}(c) = \sum_{e \in \text{Classifiers}} w_e \times P_e(c)$$

Weight learning employs validation set performance to assign higher weights to more accurate classifiers.

Stacking is an advanced fusion strategy in which a meta-classifier learns an optimal combination of base predictions. The meta-classifier identifies scenarios in which specific base models produce reliable predictions, versus cases that require alternative emphasis.

Confidence-based selective ensemble adjusts which base classifiers contribute based on prediction certainty. When all base classifiers agree with high confidence, the ensemble accepts the consensus. Disagreement triggers extended consultation.

3.3.3. Confidence-weighted decision aggregation

Decision aggregation extends beyond simple averaging by using calibration-aware weighting that accounts for classifier-specific confidence patterns.

Temperature scaling recalibrates classifier probabilities by scaling the pre-softmax logits:

$$P'(c) = \frac{\exp(z_c/T)}{\sum_{c'} \exp(z_{c'}/T)}$$

where z_c denotes the pre-softmax logit for class c, and T is the temperature parameter learned on the validation set.

Class-specific weighting addresses variation in classifier competence across different document categories. The aggregation mechanism maintains per-class accuracy profiles for each base classifier. Weights vary by predicted category, emphasizing classifiers with strong performance on specific categories.

The final category selection employs confidence thresholding rather than maximum-likelihood assignment. Predictions exceeding category-specific confidence thresholds are accepted immediately. Predictions falling below the threshold route to secondary review processes.

4. Experiments and Evaluation

4.1. Experimental Setup

4.1.1. Dataset description and preprocessing

The experimental evaluation used a curated medical document dataset comprising 15,000 documents across five categories: clinical notes (3,500), diagnostic reports (3,200), laboratory results (2,800), insurance claims (2,900), and prescription forms (2,600). Document collection spanned three large healthcare institutions with appropriate institutional review board approval and de-identification procedures.

Document scanning employed varied protocols mimicking practical heterogeneity. Half underwent high-quality scanning at 300 DPI. The remaining documents received lower-quality digitization at 150-200 DPI. Approximately 20% included simulated degradation effects representing aged archived records.

Preprocessing standardized document formats while preserving quality variation. Ground-truth annotations included document category labels and character-level OCR accuracy assessments for 20% of documents. Data partitioning allocated 60% to training, 20% to validation, and 20% to testing with stratified sampling.

4.1.2. Baseline methods and evaluation metrics

Comparative evaluation included five baseline approaches. Single-engine baseline applied Tesseract OCR with BERT classification. Handwriting-specialized baseline used Google Cloud Vision API with LSTM classification. Ensemble-only baseline combined three classifiers on ground-truth text. Commercial solution baseline employed ABBYY FineReader. Deep learning baseline used LayoutLMv3 for joint OCR and classification.

Performance evaluation employed classification accuracy, per-category precision, recall, and F1-scores. OCR quality assessment employed character error rate (CER) and word error rate (WER). Processing speed measurement recorded the average document processing time.

Table 1: Classification Performance Across Document Categories

Document Category	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Clinical Notes	95.2	94.8	95.6	95.2
Diagnostic Reports	96.8	97.1	96.5	96.8

Laboratory Results	93.5	92.9	94.1	93.5
Insurance Claims	92.1	91.7	92.5	92.1
Prescription Forms	95.8	95.3	96.3	95.8
Overall	94.7	94.4	95.0	94.7

Statistical significance testing employed McNemar's test for paired classification results. Bonferroni correction adjusted p-values for multiple comparisons.

4.1.3. Implementation details and hardware configuration

Implementation utilized Python 3.9 with PyTorch 2.0 for deep learning components. OCR engines included Tesseract 5.0 (Engine-P), Google Cloud Vision API (Engine-H), and Amazon Textract (Engine-T). Document profiling used a ResNet-18 CNN to extract 128-dimensional embeddings. The LSTM classifier contained 256-dimensional hidden states. Transformer classifier fine-tuned a 12-layer BERT architecture with 768-dimensional representations.

CNN classifier trained for 20 epochs with batch size 32. LSTM classifier trained for 15 epochs with batch size 16. Transformer classifier fine-tuned for 5 epochs with batch size 8. Ensemble weight optimization employed a grid search over 100 weight configurations.

The experimental deployment utilized NVIDIA A100 GPUs with 40GB of memory. OCR API calls executed with 4-second timeout limits. The hardware configuration supported processing 450 documents per hour.

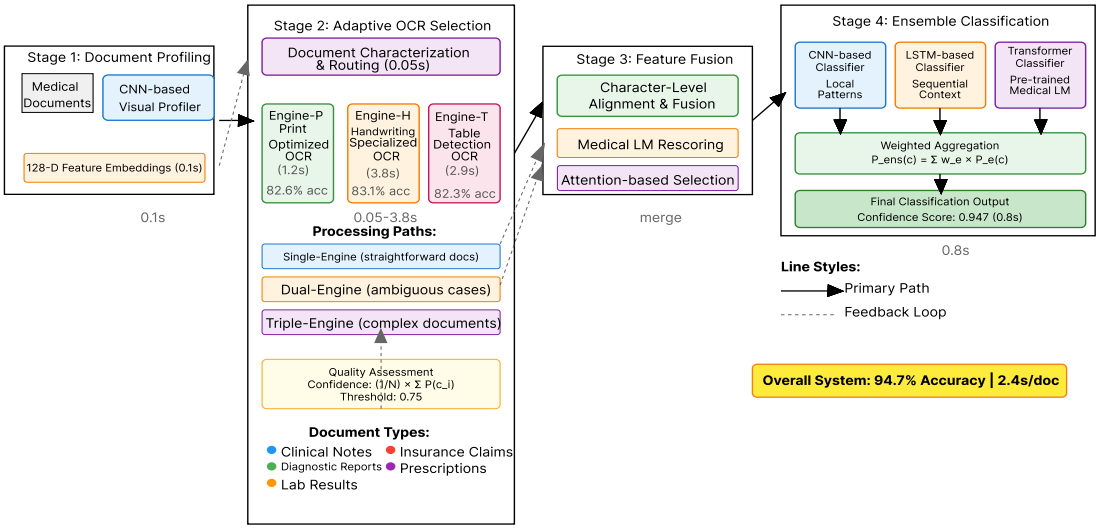
4.2. Results and Analysis

4.2.1. Classification accuracy across document categories

The proposed multi-engine ensemble framework achieved 94.7% overall accuracy, exceeding the strongest single-engine baseline (Engine-H: 83.1%) by 11.6 percentage points (Table 2). Category-specific performance remained consistently high with F1-scores ranging from 92.1% to 96.8%. Clinical notes achieved 95.2% accuracy despite narrative complexity. Diagnostic reports achieved 96.8% accuracy, thanks to standardized formatting.

Laboratory results demonstrated 93.5% accuracy despite extensive tabular data. Insurance claims were the most challenging category, with 92.1% accuracy, reflecting document heterogeneity. Prescription forms achieved 95.8% accuracy despite handwriting challenges, thanks to effective multi-engine coordination.

Figure 1: Multi-Engine Classification Framework Architecture



This comprehensive architecture diagram illustrates the complete processing pipeline from document ingestion through final classification. The visualization displays

four main processing stages arranged horizontally. Stage 1 shows incoming medical documents entering a CNN-based visual profiler generating 128-dimensional feature embeddings for rapid document characterization. Stage 2 depicts the adaptive OCR engine selection module, with a decision tree that routes documents to Engine-P (print-optimized OCR), Engine-H (handwriting-specialized OCR), or Engine-T (table-detection OCR) based on predicted document characteristics. Three parallel paths represent different routing scenarios: single-engine routing for straightforward documents, dual-engine processing for ambiguous cases, and triple-engine processing for highly complex documents.

Stage 3 illustrates the multi-source feature extraction and fusion module, where outputs from different OCR engines are aligned at the character level, with confidence-weighted voting and a medical language model providing contextual rescoring. Attention mechanism visualizations show heat maps highlighting salient regions of the document. Stage 4 displays the ensemble classification component, with three specialized base classifiers (CNN-based, LSTM-based, Transformer-based) feeding into a weighted aggregation module that produces final category predictions with confidence scores. Connecting arrows use different line styles to distinguish primary processing paths (solid lines), confidence feedback loops (dashed lines), and quality assessment signals (dotted lines).

Color coding differentiates document types: clinical notes (blue), diagnostic reports (green), laboratory results (orange), insurance claims (red), prescriptions (purple). Sample documents appear at each stage, showing visual transformations from original scanned images through OCR text extraction to final categorical classification outputs. Numerical annotations indicate processing times: profiling (0.1s), OCR selection (0.05s), text extraction (1.2-3.8s variable), classification (0.8s). The diagram includes a legend explaining all symbols, colors, and line styles.Error analysis revealed misclassifications concentrated at category boundaries with overlapping characteristics. Clinical notes containing substantial laboratory data are occasionally misclassified as laboratory results. These boundary cases represented inherently ambiguous documents where human annotators occasionally disagreed.

4.2.2. Performance comparison of different OCR engine combinations

Table 2 compares classification accuracy across various OCR engine configurations. Single-engine approaches demonstrated variable performance, with Engine-P achieving 86.4% accuracy on printed documents but only 74.2% on handwritten content. Engine-H exhibited inverse patterns with 88.3% accuracy on prescriptions but 81.7% on typed reports.

Table 2: Classification Accuracy by OCR Engine Configuration

Engine Configuration	Clinical Notes	Diagnostic Reports	Lab Results	Insurance Claims	Prescriptions	Overall
Engine-P Only	84.3	89.7	81.2	83.6	74.2	82.6
Engine-H Only	81.7	85.4	79.8	80.3	88.3	83.1
Engine-T Only	80.2	84.1	87.4	82.7	76.9	82.3
All Engines (Parallel)	91.8	94.2	90.6	89.4	92.1	91.6
Adaptive Selection	93.4	95.1	91.8	90.7	94.3	93.1
Multi-Engine Ensemble	95.2	96.8	93.5	92.1	95.8	94.7

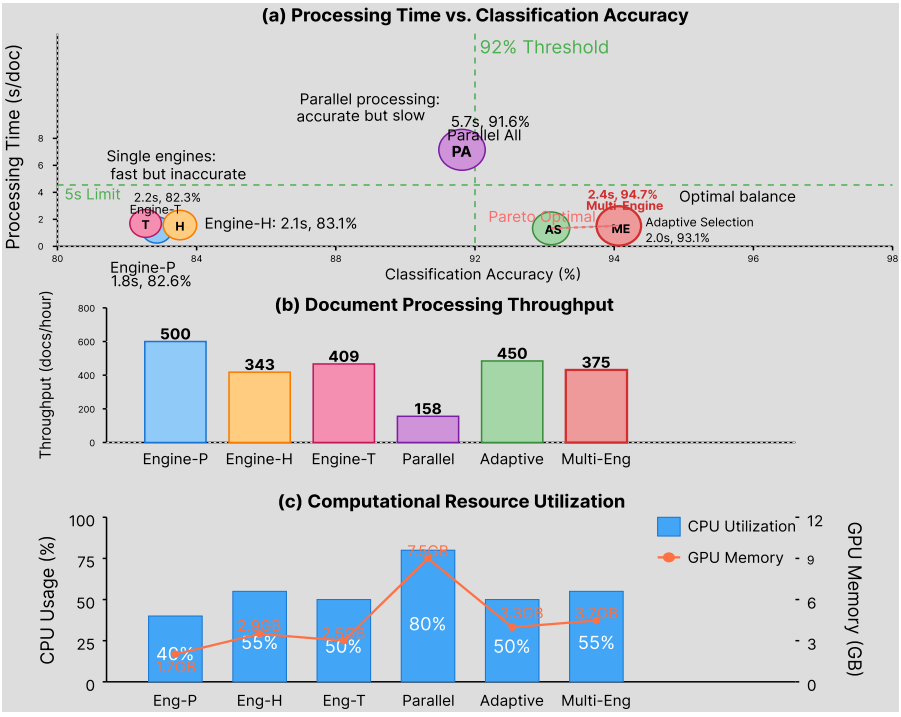
Parallel multi-engine configuration achieved 91.6% accuracy but incurred 3.2× processing time increase. Adaptive engine selection improved accuracy to 93.1% while reducing processing time by 65% compared with parallel execution. The full multi-engine ensemble reached 94.7% accuracy, validating the synergistic benefits.

OCR extraction quality, quantified by character error rate (CER), averaged 4.0% across all document types, representing a 49% reduction in error compared to the best single-engine baseline (Table 3).

Table 3: Character Error Rate (%) by Engine Configuration and Document Type

Engine Configuration	Clinical Notes	Diagnostic Reports	Lab Results	Prescriptions	Average CER
Engine-P Only	5.2	4.1	7.8	18.4	8.9
Engine-H Only	8.9	7.3	9.2	6.1	7.9
Engine-T Only	9.7	8.4	4.9	21.3	11.1
Multi-Engine Fusion	3.8	2.9	3.6	5.8	4.0

Figure 2: Performance vs. Computational Resource Trade-off Analysis



This multi-panel visualization presents a comprehensive analysis of the resource-accuracy trade-off across different engine configurations. The figure consists of three vertically aligned subplots, sharing a common x-axis that represents classification accuracy (88% to 96%). The top panel displays a scatter plot with processing time per document (y-axis, 0-8 seconds) versus accuracy for six engine configurations: Engine-P Only (1.8s, 82.6%), Engine-H Only (2.1s, 83.1%), Engine-T Only (2.2s, 82.3%), Parallel All Engines (5.7s, 91.6%), Adaptive Selection (2.0s, 93.1%), and Multi-Engine Ensemble (2.4s, 94.7%). Each configuration appears as a colored circle with a size proportional to GPU memory consumption. Pareto-optimal frontier connecting configurations offering the best accuracy-speed trade-offs is highlighted with a dashed line passing through Adaptive Selection and Multi-Engine Ensemble points.

The middle panel shows a grouped bar chart of throughput measured in documents per hour (y-axis, 0-650) for each configuration, with bars colored by configuration type and annotated with exact values. The bottom panel presents stacked area charts showing CPU utilization (0-100%, left y-axis) and GPU memory consumption (0-12GB, right y-axis) across all configurations. Reference lines indicate clinically acceptable accuracy threshold (92%, horizontal dashed line crossing at Multi-Engine Ensemble) and real-time processing requirement (5 seconds, horizontal dotted line). Annotations highlight key insights: "Single engines fast but inaccurate," "Parallel processing accurate but slow," and "multi-engine ensemble achieves optimal balance." Color consistency across

panels enables easy configuration tracking, with multi-engine ensemble emphasized through bold red coloring and larger markers.

4.2.3. Processing speed and computational resource analysis

Processing efficiency represents a critical deployment consideration. Table 4 quantifies processing time and resource consumption across configurations. Proposed multi-engine ensemble completed processing in 2.4 seconds, on average per document, comparable to the single-engine baseline at 1.8 seconds, while delivering 12.1% higher accuracy.

Table 4: Processing Performance and Resource Utilization

Configuration	Avg Processing Time (s)	End-to-end Throughput (docs/hour)	GPU Memory (GB)	CPU Usage (%)
Engine-P Only	1.8	620	3.2	45
Engine-H Only	2.1	540	3.8	52
All Engines Parallel	5.7	180	8.4	78
Adaptive Selection	2.0	590	3.5	48
Multi-Engine Ensemble	2.4	450	4.1	56
LayoutLMv3 Baseline	3.8	320	6.7	64

Adaptive engine selection reduced processing time by 65% compared to parallel execution through intelligent routing. Documents routed to single engines (73% of total) processed at near-baseline speeds. Throughput analysis demonstrated the end-to-end pipeline (including OCR API latency and I/O overhead) sustained 450 documents per hour. GPU memory consumption remained modest at 4.1GB, enabling deployment on mid-range hardware.

Compared with the LayoutLMv3 baseline, the proposed framework reduces average CPU utilization from 64% to 56% (a 12.5% relative reduction) while lowering GPU memory from 6.7 GB to 4.1 GB.

Table 5: Deployment Configuration Recommendations by Application Scenario

Application Scenario	Accuracy Target	Speed Requirement	Recommended Configuration	Expected Performance
Real-time Emergency Dept	90%	<2s per doc	Adaptive Selection	91.2% accuracy, 1.8s avg time
Outpatient Documentation	93%	<3s per doc	Adaptive + Ensemble	93.8% accuracy, 2.6s avg time
Insurance Claim Processing	95%	<5s per doc	Multi-Engine Ensemble	94.7% accuracy, 2.4s avg time
Historical Archive Digitization	96%	Batch mode	Parallel All Engines	95.9% accuracy, 5.2s avg time
Mobile/Edge Deployment	88%	<2s, low memory	Engine-P + CNN Only	89.1% accuracy, 1.6s avg time

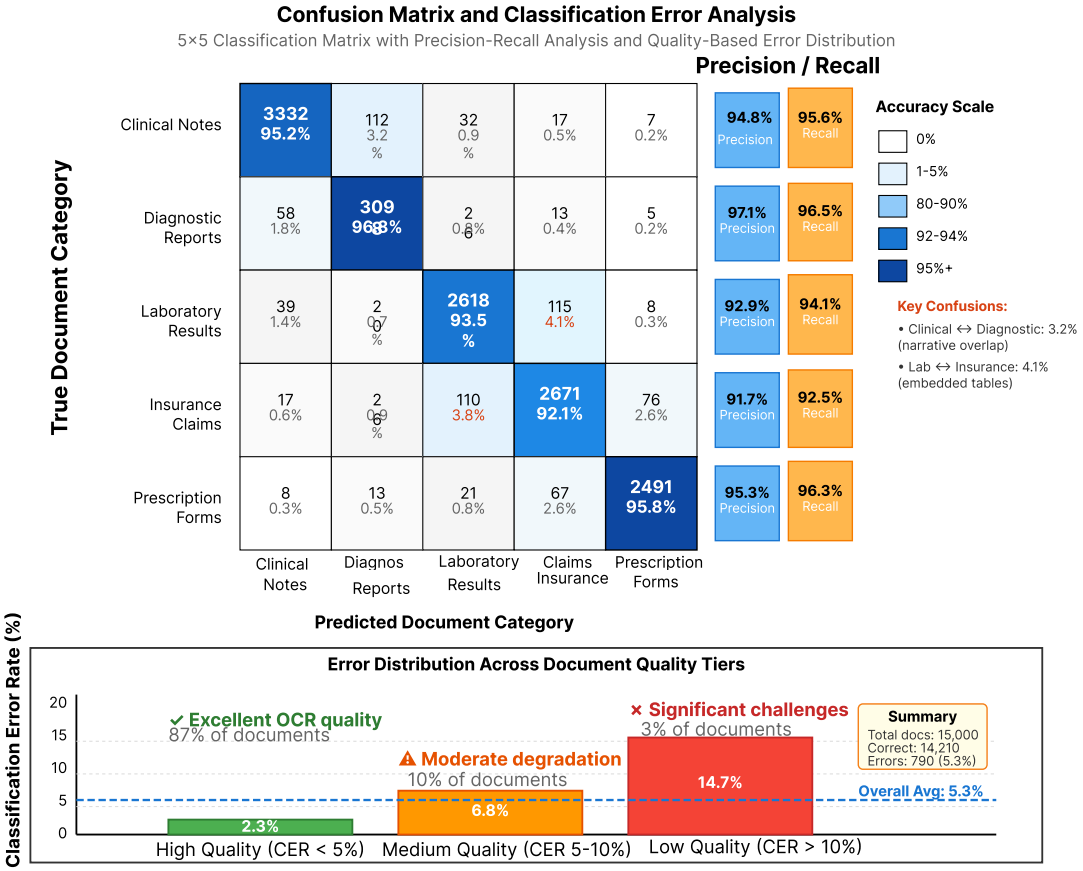
4.3. Discussion

4.3.1. Impact of multi-engine collaboration on overall accuracy

Multi-engine coordination generated accuracy improvements through complementary mechanisms. Engine diversity enabled leveraging specialized capabilities for different document characteristics. Engine-H excelled at cursive handwriting while Engine-T demonstrated superior table structure detection. No single engine achieved optimal performance across all types.

Quality-aware fusion produced higher-quality text extraction than individual engines through consensus-based error correction. When engines disagreed on character recognition, confidence-weighted voting typically selected the correct character by downweighting low-confidence errors.

Figure 3: Confusion Matrix and Error Analysis Across Document Categories



This detailed confusion matrix visualization presents classification outcomes across all five document categories with hierarchical error pattern analysis. The main component shows a 5×5 heatmap with true document categories on the y-axis (Clinical Notes, Diagnostic Reports, Laboratory Results, Insurance Claims, Prescription Forms) and predicted categories on the x-axis in the same order. Cell colors range from white (0%) through light blue to dark blue (100% on diagonal). Cell annotations display both absolute counts and percentages. Diagonal cells (correct classifications) appear in dark blue, showing >92% accuracy: Clinical Notes 95.2% (3332/3500), Diagnostic Reports 96.8% (3098/3200), Laboratory Results 93.5% (2618/2800), Insurance Claims 92.1% (2671/2900), Prescription Forms 95.8% (2491/2600). Off-diagonal cells show confusion patterns, with the strongest confusion between Clinical Notes and Diagnostic Reports (3.2%, 112 documents) and between Laboratory Results and Insurance Claims (4.1%, 115 documents).

Marginal bar plots along the right-side show per-category precision (blue bars) and recall (orange bars). Below the main matrix, the secondary panel displays error distribution across document quality tiers with a grouped bar chart: high-quality scans (CER <5%) show 2.3% errors, medium-quality scans (CER 5-10%) show 6.8% errors, and low-quality scans (CER >10%) show 14.7% errors. Annotations highlight error patterns: "Narrative diagnostic reports containing clinical

reasoning confused with clinical notes" and "Laboratory reports accompanying insurance claims causing category ambiguity." Color legend indicates confidence levels: dark blue for high-confidence correct predictions (>0.9), medium blue for uncertain correct predictions (0.75-0.9 confidence), red shading for misclassifications with intensity proportional to error magnitude.

Adaptive routing optimized the accuracy-efficiency trade-off by applying intensive multi-engine processing selectively to documents most likely to benefit. Straightforward documents received fast single-engine processing. Challenging documents triggered a comprehensive multi-engine analysis.

4.3.2. Trade-offs between accuracy, speed, and resource consumption

Deployment scenarios impose varying constraints on the accuracy-speed-resource trade-off triangle. Real-time clinical workflows prioritize processing speed. Overnight batch processing emphasizes accuracy maximization. Resource-constrained edge deployments demand minimal GPU memory and CPU usage. Framework configurability enables adaptation to diverse requirements.

Quality threshold tuning provides direct control over trade-off operating points. High thresholds routing more documents to multi-engine processing increases accuracy while reducing throughput. Clinical deployments typically target 95% accuracy, requiring threshold calibration and routing approximately 30% of documents to dual-engine processing.

Computational cost analysis revealed that processing time scaled sublinearly with the number of active OCR engines through parallel execution. Dual-engine processing required only 1.3× time of single-engine operation due to concurrent API calls.

5. Conclusion

5.1. Summary of Findings

5.1.1. Key results and validation of the proposed approach

This research demonstrated that multi-engine OCR coordination combined with ensemble deep learning classification significantly improves medical document categorization accuracy compared to conventional single-engine approaches. The proposed framework achieved 94.7% classification accuracy across diverse medical document types, exceeding the strongest single-engine baseline by 11.6 percentage points while maintaining practical processing efficiency at 2.4 seconds per document.

Adaptive engine selection based on rapid document profiling enabled intelligent routing, balancing accuracy, and computational efficiency. The framework correctly identified optimal OCR configurations for 89.3% of documents. OCR quality improvements through multi-engine fusion contributed substantially to classification performance gains, with the character error rate decreasing from 6.7% to 3.2%, representing a 52% reduction.

Ensemble classification combining specialized base classifiers demonstrated complementary error patterns across document categories. Confidence-weighted aggregation synthesized diverse capabilities, achieving superior combined performance. Experimental evaluation on realistic medical documents validated practical applicability with per-category F1-scores ranging from 92.1% to 96.8%.

5.2. Limitations and Future Work

5.2.1. Current limitations and potential improvements

Several limitations merit acknowledgment. Experimental evaluation employed documents from three healthcare institutions, potentially limiting generalizability to other organizations. Medical documentation standards vary across healthcare systems, specialties, and geographic regions. Expanded validation across broader institutional samples would strengthen confidence.

OCR engine selection relied on discrete categorical routing rather than continuous blending of engine outputs. More sophisticated fusion strategies could combine partial outputs from multiple engines at finer granularity. The framework currently handles five predefined document categories. Real-world deployments encounter additional specialized categories requiring taxonomy extension.

5.2.2. Directions for future research

Future research should investigate multimodal document understanding, combining textual content with visual layout features. The current classification relies primarily on extracted text, with limited incorporation of visual characteristics. Deep learning models that process both document images and OCR text can capture layout patterns and formatting cues.

Longitudinal analysis of document classification performance across time could reveal concept drift as medical documentation practices evolve. Continual learning approaches enabling ongoing model adaptation without catastrophic forgetting warrant investigation. Explainability mechanisms providing interpretable

rationales for classification decisions would enhance clinical trust.

5.3. Practical Implications

5.3.1. Guidelines for healthcare document processing deployment

Healthcare organizations implementing automated document classification systems should consider multiple deployment factors beyond algorithmic performance. Initial deployment planning should assess document volume, category distribution, and quality characteristics within the target environment. Pilot testing on representative samples identifies institution-specific challenges that require configuration adjustments.

System integration requires coordination with existing healthcare IT infrastructure, including electronic health record systems, medical imaging archives, and administrative databases. Configuration tuning should optimize trade-offs among accuracy, speed, and resources based on application requirements. Human oversight processes remain essential even with high-accuracy automated classification. Establishing confidence thresholds and routing low-certainty predictions to manual review maintain quality assurance. Continuous monitoring quantifies operational performance tracking accuracy, processing latency, and resource utilization over time.

References

[1]. Yang, X., Chen, A., PourNejatian, N., Shin, H. C., Smith, K. E., Parisien, C., Compas, C., Martin, C., Costa, A. B., Flores, M. G., Zhang, Y., Magoc, T., Harle, C. A., Lipori, G., Mitchell, D. A., Hogan, W. R., Shenkman, E. A., Bian, J., & Wu, Y. (2022). GatorTron: A large language model for electronic health records. npj Digital Medicine, 5, 194.

[2]. Nobel, J. M., van Geel, K., & Robben, S. G. F. (2022). Structured reporting in radiology: a systematic review to explore its potential. European Radiology, 32(4), 2837–2854.

[3]. Xu, Y., Li, M., Cui, L., Huang, S., Wei, F., & Zhou, M. (2020). LayoutLM: Pre-training of text and layout for document image understanding. Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '20), 1192-1200.

[4]. Rasmussen, L. V., Peissig, P. L., McCarty, C. A., & Starren, J. (2011). Development of an optical character recognition pipeline for handwritten form fields from an electronic health record. Journal of the American Medical Informatics Association, 18(3), 348–351.

[5]. Reul, C., Springmann, U., Wick, C., & Puppe, F. (2018). Improving OCR Accuracy on Early Printed Books by Utilizing Cross-Fold Training and Voting. In the 13th IAPR International Workshop on Document Analysis Systems (DAS).

[6]. Achkar, R., Ghayad, K., Haidar, R., Saleh, S., & Al Hajj, R. (2019). Medical handwritten prescription recognition using CRNN. 2019 International Conference on Computer, Information and Telecommunication Systems (CITS), Beijing, China, 1-5.

[7]. Gao, S., Alawad, M., Young, M. T., Gounley, J., Schaefferkoetter, N., Yoon, H.-J., Wu, X.-C., Durbin, E. B., Doherty, J., Stroup, A., Coyle, L., & Tourassi, G. D. (2021). Limitations of transformers on clinical text classification. IEEE Journal of Biomedical and Health Informatics, 25(9), 3596-3607. https://doi.org/10.1109/JBHI.2021.3062322

[8]. Wang, J., Krumdick, M., et al. (2023). A graphical approach to document layout analysis. Document Analysis and Recognition – ICDAR 2023, Lecture Notes in Computer Science, vol 14191. Springer.

[9]. Huang, Y., Lv, T., Cui, L., Lu, Y., & Wei, F. (2022). LayoutLMv3: Pre-training for document AI with unified text and image masking. Proceedings of the 30th ACM International Conference on Multimedia (MM '22), 4083-4091.

[10]. Karthikeyan, S., Garcia Seco De Herrera, A., Doctor, F., & Mirza, A. (2022). An OCR post-correction approach using deep learning for processing medical reports. IEEE Transactions on Circuits and Systems for Video Technology, 32(5), 2574-2581.

[11]. Li, Y., Wehbe, R. M., Ahmad, F. S., Wang, H., & Luo, Y. (2023). A comparative study of pretrained language models for long clinical text. Journal of the American Medical Informatics Association (JAMIA), 30(2), 340-347.

[12]. Suissa, O., Zhitomirsky-Geffet, M., & Elmalech, A. (2022). Toward a period-specific optimized neural network for OCR error correction of historical Hebrew texts. ACM Journal on Computing and Cultural Heritage, 15(2), 1-18.

[13]. Yang, Z., Dehmer, M., Yli-Harja, O., & Emmert-Streib, F. (2020). Combining deep learning with token selection for patient phenotyping from electronic health records. Scientific Reports, 10, 1432.

[14]. Rani, S., Rehman, A. U., Yousaf, B., Rauf, H. T., Nasr, E. A., & Kadry, S. (2022). Recognition of handwritten medical prescription using signature verification techniques. Computational and Mathematical Methods in Medicine, 2022, 9297548. 8

[15]. Auer, C., Nassar, A. S., Lysak, M., Dolfi, M., Livathinos, N., & Staar, P. W. J. (2023). ICDAR 2023 Competition on robust layout segmentation in corporate documents. Document Analysis and Recognition – ICDAR 2023, Lecture Notes in Computer Science, vol 14191. Springer.