Local LLM Models for Data Extraction: Comprehensive Analysis

The landscape of local LLM models for document processing has matured significantly in 2024-2025, offering compelling alternatives to cloud-based solutions for extracting tabular data from PDFs and Excel files. **Vision-language models (VLMs) now achieve 75-90% accuracy on business documents while running locally with modest hardware requirements**, MathWorks making them viable for payroll and enterprise applications where data privacy and regulatory compliance are critical.

Executive recommendations

For organizations processing payroll documents and tabular data locally, **Florence-2 emerges as the optimal starting point**, offering exceptional OCR capabilities with minimal resource requirements (3-4GB VRAM). For text-based processing of pre-extracted content, **DeepSeek Coder V2 provides state-of-the-art structured data extraction** with 73% accuracy on code generation benchmarks. The combination of these approaches, deployed through frameworks like Ollama or vLLM, delivers production-ready document processing pipelines that maintain data sovereignty while reducing operational costs by 20-50% compared to cloud alternatives. Label Your Data Cohorte

Vision-based models for PDF processing

Top performing models for local deployment

Florence-2 represents the breakthrough solution for local PDF processing, with Microsoft's compact vision-language model delivering exceptional performance per resource unit. The base model (0.23B parameters) requires only 3-4GB VRAM while achieving 700+ scores on OCRBench, surpassing many proprietary solutions. Medium +4 **Florence-VL enhances this further with depth-breadth fusion architecture**, achieving 84.9% accuracy on DocVQA tasks while using only 576 visual tokens versus 2,880 for competing models. arXiv Deep-diver

Qwen2-VL series provides scalable options from 2B to 72B parameters, with the 7B variant achieving state-of-the-art results among open models on document understanding benchmarks. The dynamic resolution mechanism handles variable image sizes effectively, while supporting up to 32K token contexts for long document processing. GitHub Qwen Resource requirements range from 8GB VRAM (quantized) to 14GB VRAM (full precision) for the 7B model.

LLaVA variants offer robust community support with extensive quantization options. LLaVA-1.5-7B provides solid baseline performance requiring 6-8GB VRAM with 4-bit quantization, making it accessible for consumer hardware deployments. GitHub Medium The larger 13B and 34B variants deliver enhanced capabilities for organizations with more substantial hardware budgets.

Performance benchmarks and accuracy metrics

Recent benchmarking reveals significant performance variations across document types and complexity levels. **Florence-2 consistently achieves 44.5-82.1% accuracy on DocVQA tasks**, with

particularly strong performance on OCR-heavy documents reaching 700+ on OCRBench evaluations. (Pondhouse Data) (Ultralytics) **Qwen2-VL-7B demonstrates 85%+ accuracy** on document understanding tasks, positioning it among the top performers for structured data extraction.

Table extraction performance varies significantly based on document quality and complexity. **High-quality digital PDFs achieve 85-95% accuracy**, while scanned documents typically see 60-80% accuracy depending heavily on OCR preprocessing quality. For payroll documents specifically, structured business forms show **90-98% field extraction accuracy** when preprocessing is properly implemented.

Resource optimization strategies

Quantization proves essential for consumer hardware deployment. 4-bit quantization reduces VRAM requirements by approximately 70% with minimal accuracy loss, (Analytics Vidhya) while 8-bit quantization offers 50% reduction with negligible performance impact. (Ionio) The GGUF format enables CPU+GPU hybrid inference, particularly beneficial for Apple Silicon and resource-constrained environments. (Tensorfuse +2)

Model-specific optimization recommendations include Florence-2-base with 4-bit quantization for environments under 12GB VRAM, Florence-2-large or Florence-VL for balanced performance scenarios (12-16GB VRAM), and Qwen2-VL-72B for maximum accuracy requirements exceeding 16GB VRAM availability.

Text-based models for structured extraction

Leading models for local deployment

DeepSeek Coder V2 emerges as the premier choice for structured data extraction from text content. Trained on 2 trillion tokens across 80+ programming languages, it significantly outperforms CodeLlama-34B while being substantially smaller. (github) **The 7B model matches 34B performance levels** with exceptional JSON and CSV output generation capabilities, crucial for payroll data structuring.

Microsoft Phi-3 family offers exceptional efficiency, with Phi-3 Mini (3.8B parameters) achieving 69% on MMLU benchmarks while requiring only 1.8GB memory with 4-bit quantization. (arXiv +2) This makes it particularly suitable for mobile and edge deployment scenarios where resources are severely constrained.

Mistral 7B and Nemo 12B provide excellent balanced performance, with Mistral 7B outperforming Llama 2 13B across benchmarks (Medium) (Mistral AI) while Nemo 12B offers 128K context length with quantization-aware training. Both models demonstrate strong multilingual capabilities essential for international payroll processing.

Performance analysis for structured tasks

Comprehensive benchmarking on structured data extraction reveals **DeepSeek Coder achieving 73% accuracy on HumanEval**, leading among local models. (github) For table extraction specifically, local models typically achieve 60-80% accuracy depending on complexity, with **Phi-3 Medium reaching approximately 75% accuracy** and **Mistral 7B achieving around 65%** on structured table processing tasks.

Processing speed varies significantly by deployment method. Consumer hardware (RTX 3060/4090) typically achieves 6-12 tokens/second with 7B models, (Medium) while processing time scales linearly with document size (6.28 seconds for single pages extending to 65.12 seconds for 50-page documents). (Procycons) (Label Your Data)

Quantization and deployment optimization

GGUF quantization provides optimal balance for most deployment scenarios. Q4_K_M quantization offers approximately 60% memory reduction while maintaining quality, requiring roughly 4GB for 7B models. (Inferless +2) Q8_0 quantization provides minimal quality loss with 5% size reduction, while Q6_K delivers good balance at 40% reduction. (github) (Inferless)

Alternative quantization methods include GPTQ for GPU-optimized inference and AWQ for activation-aware quantization providing faster performance than GPTQ with better generalization characteristics. (Tensorfuse +3)

Performance comparisons and benchmarking

Comprehensive evaluation metrics

The SUC (Structural Understanding Capabilities) benchmark provides standardized evaluation across seven structural understanding tasks. (Microsoft) (Microsoft) **GPT-4 achieves 72.48% overall accuracy** compared to GPT-3.5's 39.67%, (arXiv) establishing performance targets for local alternatives. HTML format consistently outperforms natural language approaches by approximately 6.76%, (Microsoft) indicating preprocessing strategy importance. (Microsoft)

Vision versus text-based approaches reveal complementary strengths. **Vision Language Models excel at maintaining document structure and spatial relationships**, achieving 90%+ accuracy on clear documents. Traditional OCR combined with text-based LLMs provides **better scalability and computational efficiency** but may struggle with complex layouts.

Domain-specific performance data

Financial document processing benchmarks show **GPT-4 achieving 81.5% overall accuracy** on financial QA tasks, with open-source SOLAR-10.7b providing the best performance among accessible models. (Medium) **Calculation tasks remain challenging across all models**, with GPT-4 achieving only 57% accuracy, (Medium) indicating the need for hybrid approaches combining LLMs with traditional computational methods.

Recent studies demonstrate 99% effective accuracy for business-critical documents when combining multiple OCR engines with LLM post-processing, though this requires more complex implementation and higher resource utilization.

Implementation frameworks and tools

Production deployment platforms

Ollama provides the optimal entry point for rapid prototyping and development with extremely user-friendly CLI interface, automatic model management, and built-in REST API server. While throughput is moderate (approximately 4 parallel requests by default), it enables quick proof-of-concept development. Tensorfuse +2

vLLM offers production-scale performance with PagedAttention memory management delivering 2.7x higher throughput than Ollama on benchmarks. arXiv The OpenAI-compatible API facilitates easy integration, though it requires more technical expertise and preferably NVIDIA GPU hardware. LMSYS Org +5

SGLang represents the emerging leader for complex multi-call workloads, featuring RadixAttention for KV cache reuse and achieving up to 5x throughput versus existing systems. Its adoption by major organizations like xAI demonstrates production readiness for enterprise deployments. GitHub +3

Document processing pipeline integration

PyMuPDF emerges as the speed champion for text and image extraction, providing fastest processing with comprehensive format support. **pdfplumber excels for precision applications** requiring detailed control and accurate table detection, particularly valuable for complex payroll document layouts.

Modern OCR integration combines multiple engines for optimal accuracy. **EasyOCR provides deep learning-powered recognition** for 80+ languages with GPU acceleration, while **PaddleOCR offers excellent multilingual support** particularly for Asian languages common in international payroll processing.

Complete processing architecture

Recommended production architecture implements a multi-stage pipeline: OCR/text extraction using specialized tools, preprocessing for format standardization, LLM processing with structured prompts, post-processing validation, and structured output generation. unstract Unstract **BetterOCR combines multiple OCR engines with LLM correction**, achieving superior accuracy through ensemble approaches. github

Docker containerization enables consistent deployment across environments, with production-ready containers supporting GPU acceleration and comprehensive dependency management. Sample Docker compose stacks facilitate rapid deployment of complete processing pipelines.

Preprocessing and uniformization strategies

PDF optimization techniques

Resolution standards significantly impact accuracy. 300 DPI represents the recommended baseline for optimal OCR accuracy, with 400-600 DPI necessary for text smaller than 8 points. **Maximum beneficial DPI reaches 600**, beyond which no improvement occurs while file sizes increase dramatically.

Image enhancement pipelines provide substantial accuracy improvements. Converting to grayscale, enhancing contrast by 2x, applying sharpening filters, and ensuring minimum resolution of 1000 pixels delivers 15-30% accuracy improvements for challenging documents. (Label Your Data)

Document type considerations require different approaches. Native PDFs enable direct text extraction before VLM processing, while scanned documents require full VLM pipelines for OCR. **Complex layouts benefit from models with strong spatial understanding** like Florence-VL or Qwen2-VL.

Excel processing optimization

Pre-extraction text processing proves highly effective for Excel data, achieving 90-98% accuracy on already structured content. **AI-powered tools like GPTExcel convert natural language to formulas** using transformer models, while enhanced Power Query provides ETL improvements with AI-driven data transformation.

Multi-format uniformization converts diverse input types into standardized formats suitable for LLM processing. **Table recognition algorithms preserve structure** during conversion, while advanced techniques maintain relationships between related data elements.

Resource requirements and deployment costs

Hardware specifications by use case

Small-scale deployment (under 100 documents daily) requires minimal resources: 8GB RAM with consumer CPU, utilizing Phi-3 Mini Q4_K_M (2GB) through Ollama framework. (arXiv) **Monthly operational cost approaches zero** for local processing, providing significant cost advantages over cloud alternatives.

Medium-scale operations (100-1,000 documents daily) benefit from 16GB RAM with RTX 4070 class GPUs, running Mistral 7B Q6_K or DeepSeek Coder 6.7B Q4_K_M through vLLM or llama.cpp frameworks. **Hardware amortization costs range $100-300 monthly**. (Cohorte)

Large-scale deployments (1,000+ documents daily) require 32-64GB RAM with RTX 4090 or A100 hardware, supporting Llama 3.1 8B Q4_K_M or DeepSeek Coder 33B Q4_K_M with vLLM tensor parallelism. **Operational costs range $500-1,500 monthly** including hardware depreciation.

Economic analysis and ROI projections

Cost reduction metrics demonstrate substantial benefits. Organizations typically achieve 20-50% effort reduction in document processing tasks, with some implementations showing task completion time decreasing from hours to minutes. (Label Your Data) **Processing efficiency improvements of 20%** in payroll accuracy reduce error-related costs significantly.

Break-even analysis indicates 12-18 month payback periods for medium-scale deployments when compared to equivalent cloud-based processing. **Long-term savings compound** as cloud API costs are eliminated while local hardware provides multi-year value.

Recommendations for payroll data extraction

Optimal model selection strategy

For resource-constrained payroll processing, Florence-2-base with 4-bit quantization provides exceptional OCR capabilities requiring only 3-4GB VRAM. (Medium +3) **For balanced accuracy and efficiency**, Florence-VL or Qwen2-VL-7B deliver state-of-the-art performance with reasonable resource demands (8-14GB VRAM).

Text-based processing should utilize DeepSeek Coder for generating structured output scripts, while **Phi-3 variants provide excellent efficiency** for mobile and edge deployment scenarios. **Hybrid approaches combining vision and text models** achieve optimal results for complex payroll document varieties.

Implementation roadmap

Phase 1 MVP setup (Week 1) should implement Ollama with PyMuPDF document processing, EasyOCR integration, (CODE FARM) and 7B models like Llama-3.2 or Qwen2.5. (Tensorfuse) (GitHub) **Phase 2 production readiness** (Weeks 2-3) upgrades to vLLM or SGLang serving, enhanced BetterOCR accuracy, quantization implementation, and comprehensive validation pipelines. (LMSYS Org) (Cerebrium)

Phase 3 scaling and optimization (Week 4+) adds multimodal capabilities for complex documents, batch processing with continuous batching, performance monitoring systems, and specialized handling for edge cases.

Quality assurance and compliance

Automated evaluation metrics should include exact match percentages, BLEU scores for text similarity, F1 scores for precision-recall balance, and schema compliance validation. (unstract) **Dual-LLM verification** using second models for output validation, combined with rule-based checks and confidence scoring, ensures production-grade accuracy.

Regulatory compliance benefits include complete data residency control, enhanced audit trail management, reduced third-party risks, and faster adaptation to changing compliance requirements.

Multi-state and international considerations require careful attention to varying wage laws, tax obligations, and classification requirements across jurisdictions.

Future developments and strategic considerations

Emerging technology trends

Mixture-of-experts architectures are rapidly becoming standard for efficient local deployment, with models like DeepSeek-V3 demonstrating 671B parameters with only 37B active weights. **Speculative decoding and advanced inference optimization** promise significant speed improvements while maintaining accuracy levels.

Vision-language model integration continues advancing with direct image processing capabilities, while **specialized document AI architectures** designed specifically for business applications show promising development trajectories.

Strategic implementation guidance

Organizations should **start with pilot programs** focusing on specific use cases like invoice processing or payroll document handling. **Infrastructure investment** should ensure adequate hardware and security for production deployment, while **compliance-first approaches** address regulatory requirements from project initiation.

Integration planning must consider compatibility with existing business systems, while **internal expertise development** builds teams combining AI and domain-specific knowledge. **Success metrics should balance accuracy, speed, cost-effectiveness, and compliance requirements** to ensure sustainable long-term value.

The convergence of advanced vision-language models, efficient quantization techniques, and mature deployment frameworks has created unprecedented opportunities for local document processing solutions. Infralovers Organizations implementing these technologies strategically, with proper attention to compliance, integration, and user experience, will achieve significant competitive advantages through improved cost structures, enhanced security, and superior operational efficiency. Medium