Document Layout Detection for Enhanced Retrieval in RAG Systems

Varadrajan Kunsavalikar

13 March 2025

Abstract

Document layout detection is a critical component in enhancing the performance of retrieval-augmented generation (RAG) systems. By accurately identifying structural elements such as **paragraphs, tables, headers, and metadata**, these components can be extracted, embedded, and processed separately to improve retrieval relevance. This structured approach enables **context-aware retrieval**, ensuring that different content types receive appropriate weightage based on their significance.

This paper explores multiple **document layout detection techniques**, including **YOLO-based object detection, Mistral OCR, and NVIDIA NeMo Retriever-Parse**, assessing their strengths, limitations, and suitability for various use cases. Additionally, we analyze how **layout-aware retrieval** can enhance precision by reducing noise and prioritizing structured content. Finally, we discuss future directions, including **metadata-driven retrieval strategies and layout-based chunking**, which aim to further optimize document understanding in RAG-based systems.

1 Introduction

Retrieval-Augmented Generation (RAG) systems rely on high-quality document retrieval to generate accurate and contextually relevant responses. However, traditional retrieval methods often treat documents as **unstructured text**, leading to suboptimal performance when processing structured information such as **tables, lists, and metadata**. This limitation reduces retrieval precision, as important contextual cues embedded within structured elements are often ignored.

To address this challenge, **document layout detection** enables a structured approach by:

- **Identifying distinct content types:** Detecting paragraphs, tables, headers, lists, and metadata to differentiate between structured and unstructured content.
- **Assigning separate embeddings:** Generating embeddings specific to each detected class to improve retrieval relevance.
- **Enhancing retrieval accuracy:** Prioritizing structured content and weighting different document elements appropriately.

This paper explores multiple **document layout detection techniques**, including:

- **YOLO-based detection:** Using object detection models to segment structural elements within scanned and digital documents.
- **Mistral OCR for Markdown Extraction:** Extracting text structure via markdown formatting to differentiate between headings, paragraphs, and tables.
- **NVIDIA NeMo Retriever-Parse:** Employing deep learning-based parsing to classify document elements and enhance structured retrieval.

While **document layout-based chunking** has not yet been fully integrated into our retrieval pipeline, preliminary experiments demonstrate its potential in improving retrieval effectiveness. This paper outlines our findings, evaluates various detection models, and discusses future directions for integrating layout-based retrieval strategies into RAG systems.

2 Document Layout Detection Techniques

Several methods can be used to detect the structure of a document:

2.1 YOLO-Based Layout Detection

YOLO (You Only Look Once) is a deep learning-based object detection model that can be adapted for document layout analysis. Instead of treating documents as plain text, YOLO enables the segmentation of structured elements such as tables, paragraphs, figures, and headers within scanned PDFs and images.[1]

2.1.1 Approach

To utilize YOLO for document layout detection, we followed these steps:

1. **Model Selection:** We experimented with various YOLO architectures, including YOLOv8 to YOLOv12, and tested different model sizes such as nano, small, medium, large, and extra-large. These models, as provided by Ultralytics, were evaluated for their effectiveness in document layout detection.

2. **Dataset Preparation:** Currently, no custom dataset has been prepared on our side for fine-tuning. We are using existing pre-trained YOLO models for inference.

3. **Inference and Post-Processing:** The selected models were used to detect layouts in unseen documents. The bounding box coordinates were extracted and mapped to document regions.

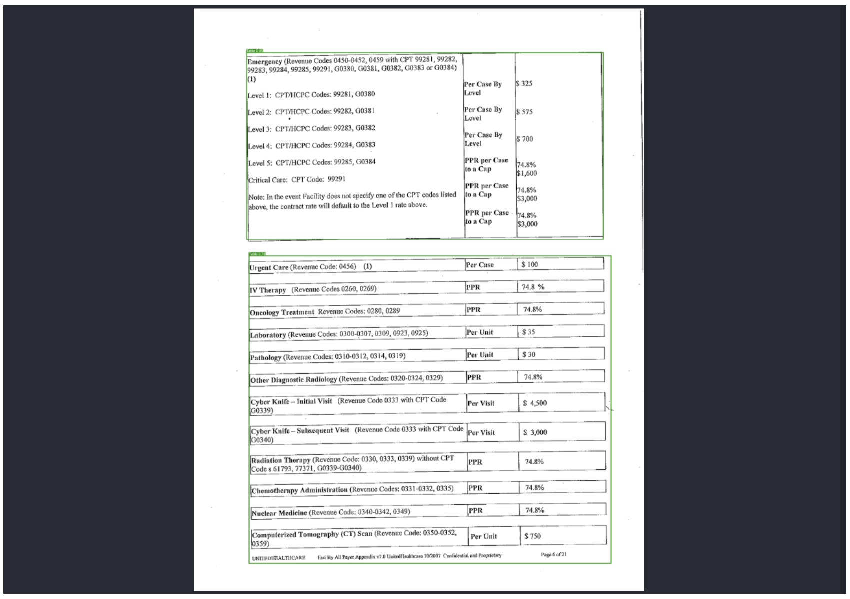
4. **Integration with Retrieval:** The detected layout structures were assigned metadata tags, enabling retrieval mechanisms to differentiate between structured and unstructured content.

5. **Future Fine-Tuning Plans:** If the results from pre-trained models are not satisfactory, we plan to fine-tune YOLO models with a labeled dataset specific to document layouts.

2.1.2 Inference and Post-Processing

The selected YOLO models were used to detect document layouts in unseen PDFs and images. The detection process identified various document components, including paragraphs, tables, and headers, along with their bounding box coordinates.

The detected bounding boxes were extracted and mapped to document regions for further processing. Figure 1 illustrates an example of detected elements in a document.



Emergency (Revenue Codes 0450-0452, 0459 with CPT 99281, 99282, 99283, 99284, 99285, 99291, G0380, G0381, G0382, G0383 or G0384) (1)		
Level 1: CPT/HCPC Codes: 99281, G0380	**Per Case By Level**	$ 325
Level 2: CPT/HCPC Codes: 99282, G0381	**Per Case By Level**	$ 575
Level 3: CPT/HCPC Codes: 99283, G0382	**Per Case By Level**	$ 700
Level 4: CPT/HCPC Codes: 99284, G0383	**PPR per Case to a Cap**	74.8% $1,600
Level 5: CPT/HCPC Codes: 99285, G0384	**PPR per Case to a Cap**	74.8% $3,000
Critical Care: CPT Code: 99291	**PPR per Case to a Cap**	74.8% $3,000
Note: In the event Facility does not specify one of the CPT codes listed above, the contract rate will default to the Level 1 rate above.		

Urgent Care (Revenue Code: 0456) (1)	**Per Case**	$ 100
IV Therapy (Revenue Codes 0260, 0269)	**PPR**	74.8 %
Oncology Treatment Revenue Codes: 0280, 0289	**PPR**	74.8%
Laboratory (Revenue Codes: 0300-0307, 0309, 0923, 0925)	**Per Unit**	$ 35
Pathology (Revenue Codes: 0310-0312, 0314, 0319)	**Per Unit**	$ 30
Other Diagnostic Radiology (Revenue Codes: 0320-0324, 0329)	**PPR**	74.8%
Cyber Knife – Initial Visit (Revenue Code 0333 with CPT Code G0339)	**Per Visit**	$ 4,500
Cyber Knife – Subsequent Visit (Revenue Code 0333 with CPT Code G0340)	**Per Visit**	$ 3,000
Radiation Therapy (Revenue Code: 0330, 0333, 0339) without CPT Code s 61793, 77371, G0339-G0340)	**PPR**	74.8%
Chemotherapy Administration (Revenue Codes: 0331-0332, 0335)	**PPR**	74.8%
Nuclear Medicine (Revenue Code: 0340-0342, 0349)	**PPR**	74.8%
Computerized Tomography (CT) Scan (Revenue Code: 0350-0352, 0359)	**Per Unit**	$ 750

UNITEDHEALTHCARE Facility All Payer Appendix v7.0 UnitedHealthcare 10/2007 Confidential and Proprietary Page 6 of 21

Figure 1: YOLO model detecting document layout components.

2.1.3 Integration with Retrieval

The detected layout structures were assigned metadata tags, enabling retrieval mechanisms to differentiate between structured and unstructured content. The following is an example JSON output of a detected document layout:

```json
    {
    "detections": [
        {
            "label": "Table",
            "confidence": 0.8997892141342163,
            "bounding_box": {
                "x_min": 296,
                "y_min": 247,
                "x_max": 2321,
                "y_max": 1309
            }
        },
        {
            "label": "Table",
            "confidence": 0.7793750166893005,
            "bounding_box": {
                "x_min": 310,
                "y_min": 1398,
                "x_max": 2365,
                "y_max": 3144
            }
        }
    ]
}
```

This structured output allows the retrieval system to:

- Extract relevant sections based on query type.
- Prioritize structured content (e.g., tables for numerical queries).
- Filter out unnecessary regions to improve retrieval precision.

2.1.4 Advantages of YOLO for Layout Detection

- **High-Speed Detection:** YOLO provides real-time detection of document structures, making it suitable for large-scale processing.
- **Scalability:** The model can be applied to diverse document formats, including scanned PDFs, structured reports, and handwritten forms.
- **Precise Bounding Box Localization:** Unlike OCR-based methods, YOLO accurately assigns spatial boundaries to each detected document component, improving segmentation quality.

2.1.5 Challenges and Limitations

- **Requirement for Annotated Training Data:** Effective training necessitates a well-labeled dataset, which may not be readily available for all document types.
- **Difficulty with Handwritten Content:** The model struggles with detecting handwritten or stylized text, making it less effective for certain document categories.
- **Computational Overhead for High-Resolution Documents:** Large multi-page PDFs require batch processing to avoid exceeding GPU memory constraints.

2.2 Mistral OCR and Markdown-Based Structure Extraction

Mistral OCR was used to extract structured information from documents by returning results in a markdown format. This approach allowed for an intuitive way to differentiate document components such as **headings, paragraphs, and tables** without relying solely on spatial positioning.[3]

2.2.1 Markdown-Based Layout Detection

The raw output from Mistral OCR was analyzed to classify elements based on markdown syntax:

- **Headings:** Identified using markdown-style notation (e.g., `# Heading 1`, `## Heading 2`).
- **Paragraphs:** Plain text lines without markdown symbols were treated as paragraph content.
- **Tables:** Recognized using markdown table structures, typically defined by pipes (`|`) and hyphens (`-`).

To process these results, a Python script was developed to parse markdown into a structured JSON format, capturing page numbers and detected elements.

2.2.2 Conversion to JSON for Structured Retrieval

The extracted markdown was transformed into JSON format to facilitate structured information retrieval. An example JSON output for a detected document page is shown below:

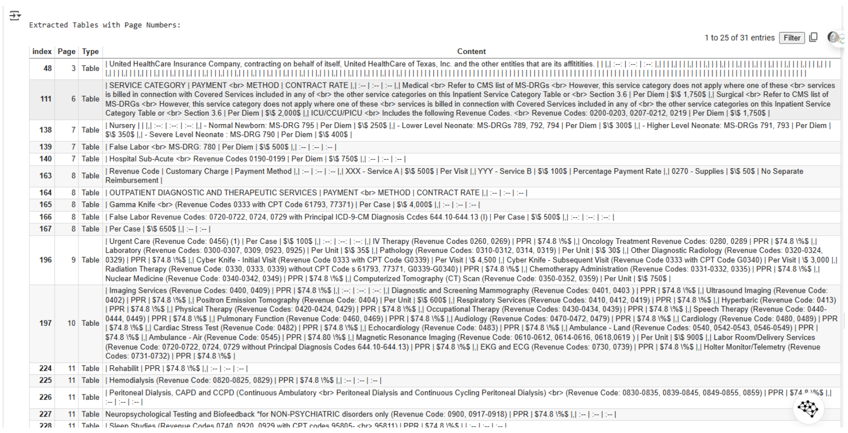
```
{
    "page": 3,
    "headings": [],
    "paragraphs": [
        "6. Effect on the Facility Agreement. The terms and conditions of the
            Facility Agreement remain in full force and effect.",
        "",
        "IN WITNESS WHEREOF, the parties hereto have executed this Agreement as
             of the Effective Date written above.",
        ""
    ],
    "tables": [
        "| United HealthCare Insurance Company, contracting on behalf of itself
            , United HealthCare of Texas, Inc. and the other entities that are
            its affiliates. |   |   |"
    ]
}
```

This structured JSON format allows retrieval systems to:

- Link extracted content to its corresponding **page number**.
- Provide targeted retrieval based on **element type** (table, paragraph, heading).
- Assign weightage to structured content during retrieval ranking.

2.2.3 Extracted Data Representation

To visualize the extracted tables and their metadata, the results were formatted into a structured DataFrame. Figure 3 presents an example of the extracted tables from a document, showcasing **page numbers, table types, and detected content**.



```
Extracted Tables with Page Numbers:
                                                                                                      1 to 25 of 31 entries  Filter
index  Page  Type   Content
48     3     Table  | United HealthCare Insurance Company, contracting on behalf of itself, United HealthCare of Texas, Inc. and the other entities that are its affititities. | | |,| :--: | :--: | :--: |,| | | |,| | | |,| | | |,| | | |,| | | |,| | | |,| | | |,| | | |,| | | |,| | | |,| | | |,| | | |,| | | |,| | | |,| | | |,| | | |,| | | |,| | | |,| | | |,| | | |,| | | |,| | | |,| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
111    6     Table  | SERVICE CATEGORY | PAYMENT <br> METHOD | CONTRACT RATE |,| :-- | :-- | :-- |,| Medical <br> Refer to CMS list of MS-DRGs <br> However, this service category does not apply where one of these <br> services is billed in connection with Covered Services included in any of <br> the other service categories on this Inpatient Service Category Table or <br> Section 3.6 | Per Diem | $\$ 1,750$ |,| Surgical <br> Refer to CMS list of MS-DRGs <br> However, this service category does not apply where one of these <br> services is billed in connection with Covered Services included in any of <br> the other service categories on this Inpatient Service Category Table or <br> Section 3.6 | Per Diem | $\$ 2,000$ |,| ICU/CCU/PICU <br> Includes the following Revenue Codes. <br> Revenue Codes: 0200-0203, 0207-0212, 0219 | Per Diem | $\$ 1,750$ |
138    7     Table  | Nursery | | |,| :--: | :--: | :--: |,| - Normal Newborn: MS-DRG 795 | Per Diem | $\$ 250$ |,| - Lower Level Neonate: MS-DRGs 789, 792, 794 | Per Diem | $\$ 300$ |,| - Higher Level Neonate: MS-DRGs 791, 793 | Per Diem | $\$ 350$ |,| - Severe Level Neonate : MS-DRG 790 | Per Diem | $\$ 400$ |
139    7     Table  | False Labor <br> MS-DRG: 780 | Per Diem | $\$ 500$ |,| :-- | :-- | :-- |
140    7     Table  | Hospital Sub-Acute <br> Revenue Codes 0190-0199 | Per Diem | $\$ 750$ |,| :-- | :-- | :-- |
163    8     Table  | Revenue Code | Customary Charge | Payment Method |,| :-- | :-- | :-- |,| XXX - Service A | $\$ 500$ | Per Visit |,| YYY - Service B | $\$ 100$ | Percentage Payment Rate |,| 0270 - Supplies | $\$ 50$ | No Separate Reimbursement |
164    8     Table  | OUTPATIENT DIAGNOSTIC AND THERAPEUTIC SERVICES | PAYMENT <br> METHOD | CONTRACT RATE |,| :-- | :-- | :-- |
165    8     Table  | Gamma Knife <br> (Revenue Codes 0333 with CPT Code 61793, 77371) | Per Case | $\$ 4,000$ |,| :-- | :-- | :-- |
166    8     Table  | False Labor Revenue Codes: 0720-0722, 0724, 0729 with Principal ICD-9-CM Diagnosis Ccdes 644.10-644.13 (I) | Per Case | $\$ 500$ |,| :--: | :--: | :--: |
167    8     Table  | Per Case | $\$ 650$ |,| :-- | :-- |
196    9     Table  | Urgent Care (Revenue Code: 0456) (1) | Per Case | $\$ 100$ |,| :--: | :--: | :--: |,| IV Therapy (Revenue Codes 0260, 0269) | PPR | $74.8 \%$ |,| Oncology Treatment Revenue Codes: 0280, 0289 | PPR | $74.8 \%$ |,| Laboratory (Revenue Codes: 0300-0307, 0309, 0923, 0925) | Per Unit | $\$ 35$ |,| Pathology (Revenue Codes: 0310-0312, 0314, 0319) | Per Unit | $\$ 30$ |,| Other Diagnostic Radiology (Revenue Codes: 0320-0324, 0329) | PPR | $74.8 \%$ |,| Cyber Knife - Initial Visit (Revenue Code 0333 with CPT Code G0339) | Per Visit | \$ 4,500 |,| Cyber Knife - Subsequent Visit (Revenue Code 0333 with CPT Code G0340) | Per Visit | \$ 3,000 |,| Radiation Therapy (Revenue Code: 0330, 0333, 0339) without CPT Code s 61793, 77371, G0339-G0340) | PPR | $74.8 \%$ |,| Chemotherapy Administration (Revenue Codes: 0331-0332, 0335) | PPR | $74.8 \%$ |,| Nuclear Medicine (Revenue Code: 0340-0342, 0349) | PPR | $74.8 \%$ |,| Computerized Tomography (CT) Scan (Revenue Code: 0350-0352, 0359) | Per Unit | $\$ 750$ |
197    10    Table  | Imaging Services (Revenue Codes: 0400, 0409) | PPR | $74.8 \%$ |,| :--: | :--: | :--: |,| Diagnostic and Screening Mammography (Revenue Codes: 0401, 0403 ) | PPR | $74.8 \%$ |,| Ultrasound Imaging (Revenue Code: 0402) | PPR | $74.8 \%$ |,| Positron Emission Tomography (Revenue Code: 0404) | Per Unit | $\$ 600$ |,| Respiratory Services (Revenue Codes: 0410, 0412, 0419) | PPR | $74.8 \%$ |,| Hyperbaric (Revenue Code: 0413) | PPR | $74.8 \%$ |,| Physical Therapy (Revenue Codes: 0420-0424, 0429) | PPR | $74.8 \%$ |,| Occupational Therapy (Revenue Codes: 0430-0434, 0439) | PPR | $74.8 \%$ |,| Speech Therapy (Revenue Code: 0440-0444, 0449) | PPR | $74.8 \%$ |,| Pulmonary Function (Revenue Code: 0460, 0469) | PPR | $74.8 \%$ |,| Audiology (Revenue Codes: 0470-0472, 0479) | PPR | $74.8 \%$ |,| Cardiology (Revenue Code: 0480, 0489) | PPR | $74.8 \%$ |,| Cardiac Stress Test (Revenue Code: 0482) | PPR | $74.8 \%$ |,| Echocardiology (Revenue Code: 0483) | PPR | $74.8 \%$ |,| Ambulance - Land (Revenue Codes: 0540, 0542-0543, 0546-0549) | PPR | $74.8 \%$ |,| Ambulance - Air (Revenue Code: 0545) | PPR | $74.80 \%$ |,| Magnetic Resonance Imaging (Revenue Code: 0610-0612, 0614-0616, 0618,0619 ) | Per Unit | $\$ 900$ |,| Labor Room/Delivery Services (Revenue Code: 0720-0722, 0724, 0729 without Principal Diagnosis Codes 644.10-644.13) | PPR | $74.8 \%$ |,| EKG and ECG (Revenue Codes: 0730, 0739) | PPR | $74.8 \%$ |,| Holter Monitor/Telemetry (Revenue Codes: 0731-0732) | PPR | $74.8 \%$ |
224    11    Table  | Rehabilit | PPR | $74.8 \%$ |,| :-- | :-- | :-- |
225    11    Table  | Hemodialysis (Revenue Code: 0820-0825, 0829) | PPR | $74.8 \%$ |,| :-- | :-- | :-- |
226    11    Table  | Peritoneal Dialysis, CAPD and CCPD (Continuous Ambulatory <br> Peritoneal Dialysis and Continuous Cycling Peritoneal Dialysis) <br> (Revenue Code: 0830-0835, 0839-0845, 0849-0855, 0859) | PPR | $74.8 \%$ |,| :-- | :-- | :-- |
227    11    Table  Neuropsychological Testing and Biofeedback *for NON-PSYCHIATRIC disorders only (Revenue Code: 0900, 0917-0918) | PPR | $74.8 \%$ |,| :-- | :-- | :-- |
228    11    Table  | Sleep Studies (Revenue Codes 0740, 0920, 0929 with CPT codes 95805- <br> 95811) | PPR | $74.8 \%$ |,| :-- | :-- | :-- |
```

Figure 2: Extracted Table DataFrame from Mistral OCR processing

2.3 NVIDIA NeMo Retriever-Parse

NVIDIA NeMo Retriever-Parse is a **general-purpose text extraction model** designed specifically for document layout analysis. Unlike traditional OCR systems, this model extracts structured text along with **bounding boxes** and corresponding **semantic classes**. This allows for better document structure understanding, which enhances retrieval pipelines.[4]

2.3.1 Capabilities and Use Case

The NeMo Retriever-Parse model is designed for:

- **Text extraction from images and PDFs**, including PowerPoint (PPT) files.
- **Classifying document elements**, such as **title, section, caption, index, footnotes, lists, tables, bibliography, and images**.
- **Generating structured outputs**, which can help improve **retriever systems and LLM/VLM training**.

The extracted text and layout metadata can be integrated into document understanding pipelines and retrieval-augmented generation (RAG) systems.

2.3.2 Model Architecture and Processing

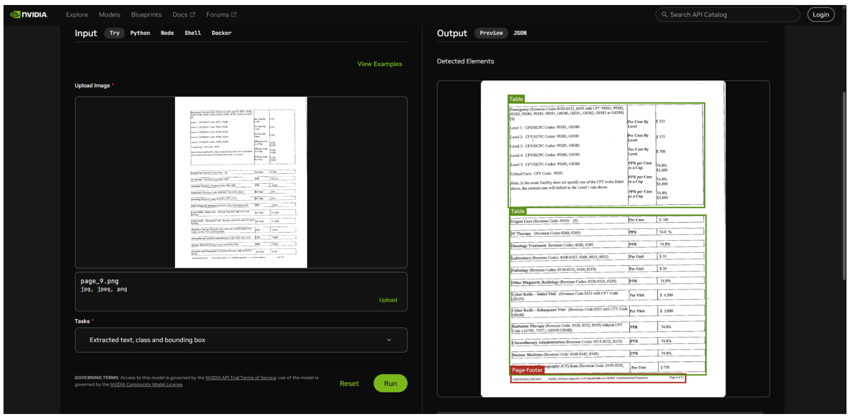
NeMo Retriever-Parse is based on a **Transformer-based vision-encoder-decoder model**. The architecture consists of:

- **Vision Encoder:** A ViT-H model that processes document images.
- **Adapter Layer:** Uses 1D convolutions and normalization techniques to reduce the latent space from 1280 tokens to 320 tokens.
- **Decoder:** Built on mBart-10 blocks for text extraction and structuring.
- **Tokenizer:** Uses the Galactica tokenizer, optimized for scientific and structured text.

The model takes in **RGB images** with a resolution ranging from **1024x1280 to 1648x2048 pixels** and outputs structured text in the form of a **formatted string containing text, bounding boxes, and class attributes**.

2.3.3 Integration and Experimentation

We experimented with NeMo Retriever-Parse to assess its performance in document layout detection. The API was used to extract structured text and classify elements from documents. A sample output of detection is shown below:



page_9.png
jpg, jpeg, png

Tasks: Extracted text, class and bounding box

GOVERNING TERMS: Access to this model is governed by the NVIDIA API Trial Terms of Service; use of the model is governed by the NVIDIA Community Model License.

Figure 3: Detected Layouts using NVIDIA model

The detected elements, including titles, tables, and paragraphs, provide additional metadata that can be used for **context-aware retrieval**.

2.3.4 Advantages of NeMo Retriever-Parse

- **Comprehensive Layout Analysis:** Extracts structured text and classifies document components.
- **Improved Metadata for Retrieval:** Bounding boxes and class labels enhance indexing and searchability.
- **Designed for LLM and VLM Training:** Helps generate structured training data for large-scale AI models.

2.3.5 Challenges and Limitations

- **High Computational Cost:** Requires NVIDIA GPUs with TensorRT-LLM for optimal performance.
- **Not Yet Production-Ready:** The model is still in a beta release and is intended for research purposes.
- **Limited Fine-Tuning Documentation:** Unlike YOLO and Mistral OCR, fine-tuning procedures for domain-specific applications are not well documented.

2.4 Comparison of Document Layout Detection Methods

Each of the evaluated models—YOLO, Mistral OCR, and NeMo Retriever-Parse—has distinct advantages and trade-offs. The choice of the model depends on factors such as **deployment feasibility, cost, accuracy, and customization needs**. Table 1 summarizes key differences.

Table 1: Comparison of Document Layout Detection Methods

Feature	**YOLO**	**Mistral OCR**	**NeMo Retriever-Parse**
Bounding Box Detection	Yes	Yes	Yes
Semantic Class Extraction	No	Limited	Yes
Handles Handwritten Text	No	Limited	Yes
Table Structure Detection	Moderate	High	High
Document Element Classification	No	Yes	Yes
Requires GPU	No	No	Yes

2.4.1 Use-Case Based Selection of Models

Selecting the right document layout detection model depends on specific **business constraints and technical feasibility**. The following scenarios outline when each model is most suitable:

- **On-Premise Deployment:** If an organization requires a fully **on-premise** solution due to **data security regulations or network restrictions**, then **YOLO** is the only viable choice. Mistral OCR and NeMo Retriever-Parse require cloud-based API access.
- **Cost Considerations:** If cost is a major factor, then **YOLO and Mistral OCR** are preferable. **YOLO** is **free** but requires extensive fine-tuning and training, whereas **Mistral OCR** provides a relatively affordable API-based solution. **NeMo Retriever-Parse** is the most costly but offers higher accuracy.
- **Accuracy vs. Cost Trade-Off:** If **accuracy is the primary concern and cost is not an issue**, then **NeMo Retriever-Parse** is the best option as it provides **structured metadata, bounding boxes, and classification of document components with high precision**. However, **Mistral OCR** is also a good alternative at a lower cost.
- **Customization and Control:** If full control over the model and data is needed (e.g., for domain-specific fine-tuning), then **YOLO** is the best option. Unlike Mistral OCR and NeMo, which are third-party APIs, **YOLO allows custom training on private datasets**. However, this requires significant effort for **data gathering, annotation, fine-tuning, and model optimization**.
- **Support for Handwritten Text:** If the use case involves handwritten documents, then **NeMo Retriever-Parse** is the best option, as **YOLO and Mistral OCR** have limited capabilities in recognizing handwritten content.
- **Real-Time Processing:** If **real-time or edge computing is required**, then **YOLO** is the preferred choice since it can be deployed on edge devices, unlike API-based solutions which depend on network latency.

2.4.2 Summary Table: Model Selection Based on Use Cases

Table 2: Model Selection Based on Deployment and Business Needs

Use Case	**YOLO**	**Mistral OCR**	**NeMo Retriever-Parse**
On-Premise Deployment	Yes	No	No
Cloud API Accessibility	No	Yes	Yes
Cost Efficiency	Free (Training Required)	Low	Expensive
High Accuracy Needed	Requires Fine-Tuning	Good	Best
Customization (Fine-Tuning Possible)	Yes	No	No
Handwritten Text Support	Limited	Limited	Yes
Real-Time Processing	Yes	No	No

2.4.3 Summary of Deployment Considerations

The final choice of the model depends on multiple factors, summarized as follows:

- **For high accuracy with structured output (metadata + bounding boxes) $\rightarrow$ NeMo Retriever-Parse**
- **For an affordable cloud-based API with good accuracy $\rightarrow$ Mistral OCR**
- **For on-premise or edge computing with full control over training $\rightarrow$ YOLO**

This comparison helps in making an informed decision based on the specific needs of the business and the technical constraints of the deployment environment.

3 Impact on Retrieval Performance

The integration of document layout detection enhances retrieval performance by embedding different document classes separately. This structured approach enables the following improvements:

- **Improved Retrieval Accuracy:** By focusing on semantically meaningful sections, retrieval precision is significantly enhanced.
- **Context-Aware Answers:** Prioritizing structured content such as tables, headers, and paragraphs allows for more relevant responses.
- **Reduced Noise in Retrieval:** By eliminating irrelevant sections, retrieval recall and precision are optimized.

Although **layout-based chunking is not yet implemented**, it remains an essential goal for future research. The ability to selectively retrieve and rank different document elements based on their **layout structure** could significantly improve retrieval efficiency.

4 Future Work

While our experiments successfully detect document layouts, integrating this metadata into retrieval pipelines presents ongoing challenges. Future research will explore:

- **Metadata-Driven Retrieval:** Developing ranking mechanisms that prioritize structured content such as tables, headings, and key-value pairs.
- **Layout-Aware Chunking Strategies:** Implementing document chunking techniques based on detected layouts to improve retrieval relevance.
- **Hybrid Model Optimization:** Exploring a unified pipeline that leverages **YOLO-based detection, Mistral OCR, and NVIDIA NeMo Retriever-Parse** for enhanced document understanding and retrieval.
- **Trying Other well known Object Detection Algorithms:** Implementing object detection algorithms like Fast R-CNN, SSD and YOLO to improve retrieval relevance.[5]

By addressing these areas, we aim to bridge the gap between **layout-aware document processing and retrieval-augmented generation (RAG)**.

5 Conclusion

Document layout detection plays a crucial role in **enhancing retrieval efficiency** in **RAG-based systems**. By leveraging layout metadata and structure-aware embeddings, we can significantly improve **precision, recall, and contextual understanding** in retrieval tasks.

This research demonstrates the potential of **hybrid retrieval techniques** that combine **semantic, lexical, and structural document representations**. However, further optimization is required to **incorporate metadata into retrieval pipelines, develop ranking strategies based on layout structures, and implement layout-aware chunking mechanisms**.

Future research will focus on **fine-tuning retrieval models** to maximize **structured content understanding** while balancing computational efficiency. The integration of **layout-aware embeddings, hybrid ranking techniques, and AI-driven retrieval models** will pave the way for more **accurate, efficient, and intelligent document processing systems**.

6 References

References

[1] YOLOv8 to YOLOv12 Available: https://docs.ultralytics.com/models/yolov8/.

[2] docyolo paper Available: https://arxiv.org/abs/2410.12628.

[3] Mistral AI, "Mistral OCR Documentation," Available: https://mistral.ai/docs/ocr.

[4] NVIDIA, "NeMo Retriever-Parse Model Documentation," Available: https://developer.nvidia.com/nemo-retriever-parse.

[5] Object Detection Algorithms. Available: https://medium.com/ibm-data-ai/faster-r-cnn-vs-yolo-vs-ssd-object-detection-algorithms-18badb0e02dc.

[6] Hugging Face, "Transformer Models Documentation," Available: https://huggingface.co/docs/transformers/en/model_doc/mbart.

[7] The Galactica Team, "Galactica: A Large Language Model for Science," Available: https://arxiv.org/abs/2211.09085.