types. This allows us to conduct targeted hard sample mining based on element types and to construct similar challenging examples through data synthesis. Finally, we incorporated manual annotation for a small number of corner cases to complete the construction of the training data.

Comprehensive benchmarking on the public benchmarks, including OmniDocBench v1.0, v1.5 [16] and olmOCR-Bench [12], and in-house ones demonstrate that PaddleOCR-VL achieves SOTA performance in document parsing task, significantly outperforming existing pipeline-based solutions and exhibiting strong competitiveness against leading vision-language models (VLMs). Moreover, PaddleOCR-VL is optimized for efficiency, delivering substantially lower latency and higher throughput than competing approaches.

PaddleOCR-VL actively addresses current challenges in document processing with a high-performance, resource-efficient multimodal document parsing solution. Its key contributions include:

- **Compact yet Powerful VLM Architecture:** We present a novel vision-language model that is specifically designed for resource-efficient inference, achieving outstanding performance in element recognition. By integrating a NaViT-style dynamic high-resolution visual encoder with the lightweight ERNIE-4.5-0.3B language model, we significantly enhance the model's recognition capabilities and decoding efficiency. This integration maintains high accuracy while reducing computational demands, making it well-suited for efficient and practical document processing applications.
- **High-quality Data Construction Methodology:** We propose a systematic and comprehensive methodology for constructing high-quality datasets, providing a solid train data foundation for efficient and robust document parsing. This methodology not only enables us to construct high-quality data on demand, but also provides a new perspective on the automated generation of high-quality data.
- **SOTA Performance Document Parsing:** PaddleOCR-VL achieves state-of-the-art performance in document parsing task. It excels in recognizing complex document elements, such as **text, tables, formulas, and charts**, making it suitable for a wide range of challenging content types, including handwritten text and historical documents. Supporting **109 languages**, including major global languages and those with diverse scripts like Russian, Arabic, and Hindi, PaddleOCR-VL is highly applicable to multilingual and globalized document processing scenarios.

2. PaddleOCR-VL

2.1. Architecture

PaddleOCR-VL decomposes the complex task of document parsing into a two stages, as illustrated in Figure 2. The first stage, PP-DocLayoutV2, is responsible for layout analysis, where it localizes semantic regions and predicts their reading order. Subsequently, the second stage, PaddleOCR-VL-0.9B, leverages these layout predictions to perform fine-grained recognition of diverse content, including text, tables, formulas, and charts. Finally, a lightweight post-processing module aggregates the outputs from both stages and formats the final document into structured Markdown and JSON.

2.1.1. Layout Analysis

Considering that end-to-end approaches based on VLM rely on long-sequence autoregressive processes, which result in high latency and memory consumption, and increase the risk of

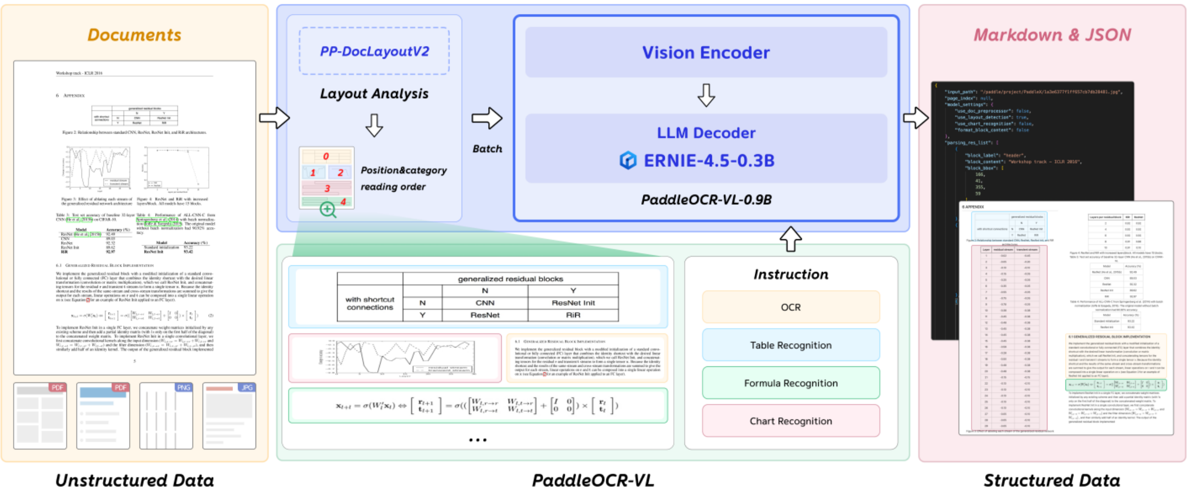

Figure 2 | The overview of PaddleOCR-VL.

unstable layout analysis and hallucinations—problems that are particularly pronounced in multi-column or mixed text–graphic layouts—we employ a dedicated lightweight model for layout analysis, focusing specifically on element detection, classification, and reading order prediction.

Specifically, we decouple the layout analysis process by introducing an independent model, PP-DocLayoutV2, dedicated solely to this task. PP-DocLayoutV2 consists of an object detection model (RT-DETR [17]) for elements localization and classification, as well as a lightweight pointer network [18] with six transformer layers to accurately predict the reading order of layout elements.

This separation enables us to fully leverage the advanced capabilities of the vision model, which typically requires lower input image resolution, and contains significantly fewer parameters. As a result, it achieves stable and accurate layout analysis, without the instability issues that may arise in end-to-end approaches.

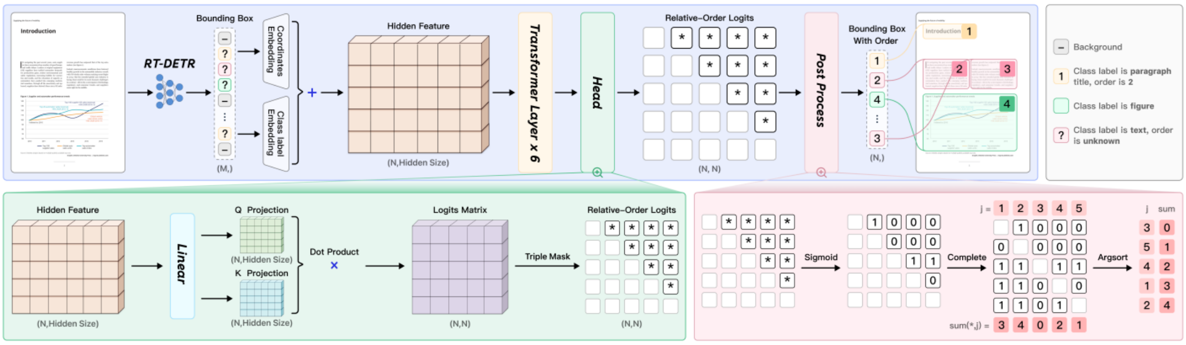

Figure 3 | Architecture of layout analysis model.

Architecturally, PP-DocLayoutV2 is composed of two sequentially connected networks, as shown in Figure 3. The first is an RT-DETR-based [17] detection model that performs layout element detection and classification. The detected bounding boxes and class labels are then passed to a subsequent pointer network, which is responsible for ordering these layout elements.

Specifically, we first apply per-class thresholds to select foreground proposals for the ordering network. The selected proposals are embedded using absolute 2D positional encodings and class label embeddings. Additionally, the encoder attention incorporates a geometric bias mechanism from Relation-DETR [18] to explicitly model pairwise geometric relationships among elements. The pairwise relation head linearly projects element representations into query and key vectors, then computes bilinear similarities to produce pairwise logits, resulting in an $N \times N$ matrix that represents the relative order between each pair of elements. Finally, a deterministic win-accumulation decoding algorithm recovers a topologically consistent reading order for the detected layout elements.

In comparison to other specialized models, such as LayoutReader [19], our model achieves higher performance with fewer parameters by efficiently extending RT-DETR [17] with a pointer network.

2.1.2. Element-level Recognition

We systematically explore architecture configurations optimized for high accuracy and low computational overhead, and propose the PaddleOCR-VL-0.9B as shown in Figure 4.

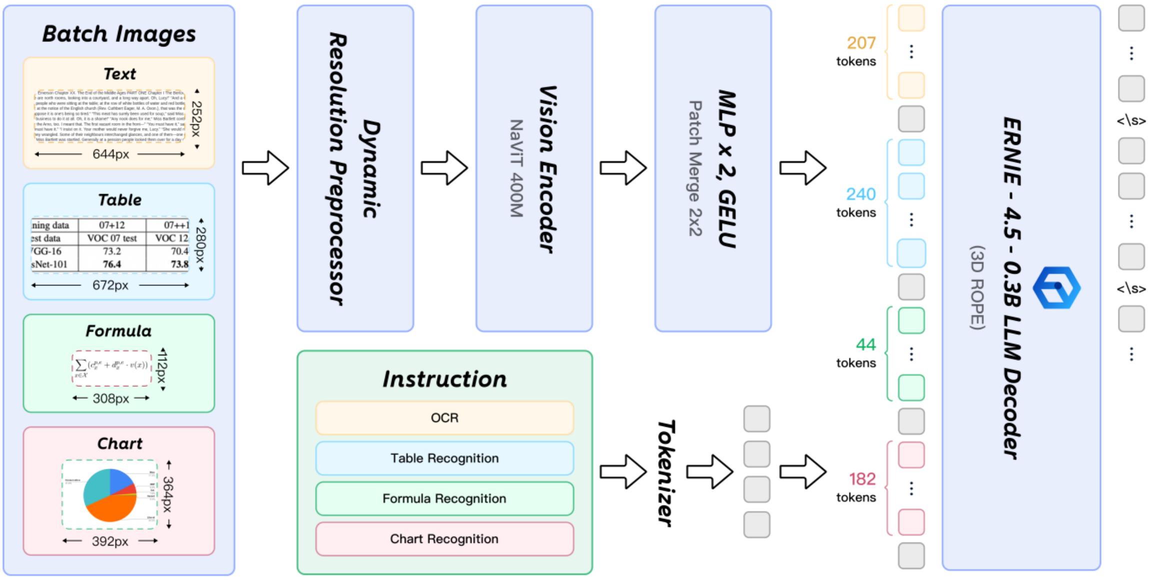

Figure 4 | Architecture of PaddleOCR-VL-0.9B.

We adopted an architectural style inspired by LLaVA [20], integrating a pre-trained vision encoder with a dynamic resolution preprocessor, a randomly initialized 2-layer MLP projector, and a pre-trained large language model. Our architecture achieves a balance the scale of vision and language models to optimize performance in multi-elements recognition tasks.

Compared to earlier document parsing models based on fixed-resolution or tiling-based approaches [4, 14, 21], our approach utilizes native dynamic high-resolution preprocessing. For the vision encoder, we employed a NaViT-style [15] encoder initialized from Keye-VL's [22] vision model, which support native-resolution inputs. This design enables the vision-language model to handle images of arbitrary resolution without distortion, yielding fewer hallucinations and stronger performance on text-intensive tasks.