DocSpiral: A Platform for Integrated Assistive Document Annotation through Human-in-the-Spiral

Qiang Sun[1*], Sirui Li[2], Tingting Bi[3], Du Huynh[1], Mark Reynolds[1], Yuanyi Luo[4], Wei Liu[1*]
[1]The University of Western Australia, Perth, WA, Australia,
[2]Murdoch University, Perth, WA, Australia,
[3]The University of Melbourne, Melborune, VIC, Australia,
[4]Sinograin Chengdu Storage Research Institute Co., Ltd., Chengdu 610091, China
[*]**Correspondence:** pascal.sun@research.uwa.edu.au, wei.liu@uwa.edu.au

Abstract

Acquiring structured data from domain-specific, image-based documents—such as scanned reports—is crucial for many downstream tasks but remains challenging due to document variability. Many of these documents exist as images rather than as machine-readable text, which requires human annotation to train automated extraction systems. We present **DocSpiral**, the first Human-in-the-Spiral assistive document annotation platform, designed to address the challenge of extracting structured information from domain-specific, image-based document collections. Our spiral design establishes an iterative cycle in which human annotations train models that progressively require less manual intervention. **DocSpiral** integrates document format normalization, comprehensive annotation interfaces, evaluation metrics dashboard, and API endpoints for the development of AI / ML models into a unified workflow. Experiments demonstrate that our framework reduces annotation time by at least 41% while showing consistent performance gains across three iterations during model training. By making this annotation platform freely accessible, we aim to lower barriers to AI/ML models development in document processing, facilitating the adoption of large language models in image-based, document-intensive fields such as geoscience and healthcare. The system is freely available at: `https://app.ai4wa.com`. The demonstration video is available: `https://app.ai4wa.com/docs/docspiral/demo`.

1 Introduction

Unstructured data are information that does not adhere to a predefined data model or format, such as free text, images, audio, video, and social media content, etc. (Nick-Barney, 2025). Unstructured data are widely believed to form 80%-90% of the world's global data assets (Heeg, 2023). Due to the sheer complexity of dealing with such data, unstructured data are often referred to as "dark data" and are significantly underutilized. To unlock the wealth of valuable knowledge hidden in unstructured data, various techniques such as knowledge graph constructions and Retrieval Augmented Generation (RAG (Stewart and Liu, 2020)) can be employed, provided that the documents are first processed to extract relevant textual data.

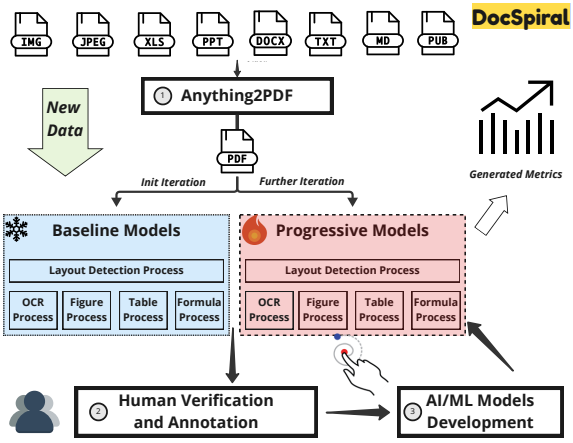

Figure 1: Our **DocSpiral** framework converts documents to PDF and processes them through iterative cycles where human verification creates annotations that improve AI/ML models, reducing effort and enhancing performance within each iteration.

Most existing document processing frameworks (Faysse et al., 2025; Shen et al., 2021; Wang et al., 2024) rely on general purpose pipelines that convert raw documents into machine-readable semi-structured formats (e.g. markdown, JSON) suitable for machine consumption. However, these pipelines face significant challenges when applied to domain-specific document collections, which often contain specialized terminology, unique layouts, and field-specific visual elements such as maps (Zhao et al., 2024a; Riedler and Langer, 2024; Fan et al., 2024). **Firstly**, traditional document processing systems often struggle to extract information accurately from such complex

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations), pages 267–274

Table 1: Comparison of Document Annotation Tools. Due to the emergence of LLM and RAG technologies still being a recent development, tools supporting figure, formula, and table understanding capabilities remain scarce. (*Ann. ⇒ Annotation, Conv. ⇒ Conversion: transforming data from image to another while preserving complete factual content without interpretation, Und. ⇒ Understanding: generating descriptive text based on a given image, involving interpretation, meaning inference, pattern recognition, and subjective judgment about data implications.)*

Tool	Reference	Open Access	Layout Ann.	OCR Ann.	Figure		Formula		Table	
					Conv.	Und.	Conv.	Und.	Conv.	Und.
ABBYY FineReader	(ABBYY, 1993)	No	×	✓	×	×	×	×	✓	✓
Transkribus	(READ-COOP SCE, 2013)	No	×	✓	×	×	×	×	✓	×
Coco Annotator	(Brooks, 2019)	Yes	✓	×	×	×	×	×	×	×
PDFAnno	(Shindo et al., 2018)	Yes	×	✓	×	×	×	×	×	×
Label Studio	(Tkachenko et al., 2020)	Partially	✓	✓	×	×	×	×	×	×
PPOCRLabelv2	(PFCCLab, 2020)	Yes	✓	✓	×	×	×	×	✓	✓
PAWLS	(Neumann et al., 2021)	Yes	✓	×	×	×	×	×	×	×
Tagtog	(TagTog team, 2023)	No	×	✓	×	×	×	×	×	×
Prodigy	(Explosion AI, 2023)	No	✓	×	×	×	×	×	×	×
Callico	(Kermorvant et al., 2024)	No	×	✓	✓	✓	×	×	×	×
DocSpiral	Ours	Yes	✓	✓	✓	✓	✓	✓	✓	✓

sources, creating barriers to knowledge utilization in fields such as geoscience, and healthcare (Zhu et al., 2024). The high variability and complexity of domain-specific documents necessitate human expertise to guide and refine automated processing systems. **Secondly**, in specialized domains, a significant portion of valuable documents exist as scanned PDFs rather than digital formats. For example, Western Australia's Mineral Exploration Reports[1], dating back to 1888, consist primarily of handwritten and printed documents that were later scanned into PDFs (Riganti et al., 2015). This poses challenges for automated processing, as these documents require layout analysis, optical character recognition (OCR), and figure/table/formula processing before they can be utilised in AI-driven applications. **Thirdly**, existing annotation tools have significant limitations for document processing tasks. Classical tools such as COCO Annotator (Brooks, 2019) were primarily designed for image annotation and lack optimization. Although PAWLS (Neumann et al., 2021) offers more specialized PDF labeling capabilities, it still suffers from rigid annotation schema. Currently, there is no comprehensive document annotation system that can efficiently support the entire document annotation pipeline. Due to the diversity of output structures for figure/table/formula processing, we also need such systems capable of addressing the dynamic complexity of these tasks through features like dynamic annotation form generation.

To address these challenges, we introduce **DocSpiral**, the first Human-in-the-Spiral assistive document annotation platform designed to facilitate domain-specfic document processing. As shown in Figure 1, our system first converts various document formats to standardized PDF format through the **Anything2PDF** module. This unified format enables integration with layout analysis models like *DocLayout-YOLO* as **Baseline Models**, which predict bounding boxes using a generic layout schema (Zhao et al., 2024d). Based on bounding-box labels, specialized downstream processing (OCR, Figure/Table/Formula processing) is triggered using the corresponding **Baseline Models**. Our human-in-the-spiral approach, supported by an interactive interface, allows experts to review, verify, and annotate model output. The annotated data are then used to train or fine-tune to obtain **Progressive Models** that better meet user requirements. Unlike existing tools such as COCO Annotator (Brooks, 2019), our system leverages pretrained models for initial annotation, significantly reducing the manual labeling workload. Users can focus primarily on corrections through an intuitive web-based interface, which leads to at least 41% time reduction, and in some cases up to 75%.

Our work makes three key contributions:

- **Comprehensive Document Annotation System** – We develop the first (as shown in Table 1) full-featured annotation system that supports the entire document processing pipeline, from layout detection and OCR to tables, figures, and formulas conversion and understanding tasks. Our flexible and customizable annotation schema design accommodates the complexity and diversity of layout, figure/formula/table processing tasks.
- **Assisted Spiral Improvement Framework** – We introduce an iterative, human-in-the-spiral approach where human verification and model

[1] https://www.dmp.wa.gov.au/WAMEX-Minerals-Exploration-1476.aspx

training reinforce each other over successive cycles. This process progressively reduces annotation effort while improving model performance.
- **Open and Deployable Solution** – We make DocSpiral freely accessible to researchers while also offering deployable solutions for organizations with privacy constraints, thereby removing barriers to LLM adoption in specialized domains.

2 Related Work

Existing annotation tools Comprehensive document annotation requires support for various tasks, including ***Layout detection*** outputs bounding boxes with category labels (content, title, figure, table, formula, footnote, etc); ***OCR*** requires accurate text transcriptions of segmented images; ***Table processing*** includes both structural conversion (to LaTeX (Xia et al., 2024), HTML (Wan et al., 2024), JSON (Ali Khandokar and Deshpande, 2025)) and semantic understanding (Zhao et al., 2024c) for RAG systems as explained in Table 1; ***Formula processing*** is similar to table processing with structural conversion typically outputting LaTeX (Xia et al., 2025); ***Figure processing*** prioritizes understanding visual elements over conversion due to representation diversity and difficulty (Zhao et al., 2024b). Tools such as LayoutParser (Shen et al., 2021), MinerU (Wang et al., 2024), and Docling (Team, 2024) provide the ability to integrate with parts or all of these specialized models to build an end-to-end pipeline; however, when errors occur, these tools lack mechanisms that allow users to fix specific problems or improve individual models at intermediate stages.

No existing annotation tool fully addresses these needs, as illustrated in Table 1, particularly for semantic understanding annotation of formulas, tables, and figures. Commercial solutions such as Tagtog (TagTog team, 2023), Callico (Kermorvant et al., 2024), and Prodigy (Explosion AI, 2023) offer partial capabilities. Although there are specialized tools for individual tasks (Huynh et al., 2022; gipplab, 2019), the research community lacks a unified system that integrates these capabilities into a cohesive pipeline that facilitates human intervention for error correction and iterative model improvement throughout the entire document processing workflow, to produce structured high-quality outputs from unreconstructed data formats.

Human role in annotation systems Traditional annotation pipelines follow a *human-off-the-loop* paradigm, where annotators exhaustively label data offline before model training or fine-tuning (Tkachenko et al., 2020; Kermorvant et al., 2024; Explosion AI, 2023; Neumann et al., 2021). While effective, this approach is labour-intensive and impractical for large or continuously evolving datasets (Wu et al., 2022; Peña et al., 2024).

Instead, our document processing framework shifts towards a *human-in-the-loop* approach (Nahavandi, 2017), using baseline models or LLMs for assistive annotations—such as figure understanding or layout detection—while humans intervene selectively for validation and correction.

Building on this, we introduce the *human-in-the-spiral* framework, where new data is first processed by prior models, then undergoes targeted expert review, followed by iterative model enhancement (as shown in Figure 1). This positive feedback loop improves model performance in an upward spiral while minimizing manual annotation.

3 System design and implementation

3.1 Requirement analysis

We present **DocSpiral**, a web-based document processing pipeline initiated from document layout detection (Huang et al., 2022). Its primary objective is to enable efficient review and annotation of model outputs, generating high-quality annotated data for iterative models improvement. This approach enhances downstream tasks, such as knowledge graph construction and RAG applications, in specialized domains like geology and healthcare, where valuable documents are often paper-based, or in scanned images with a rich mix of multimodal contents, such as photos, maps, charts, and tables.

To ensure usability and adaptability, the system minimizes human effort, incorporates automatic evaluation metrics dashboard generation, and maintains a modular structure for extensibility. It supports agile methodologies, allowing researchers to rapidly develop and refine different document processing models in response to the fast paced GenAI challenges. As shown in Figure 1, **DocSpiral** consists of three integrated modules:

- **Anything2PDF** converts diverse formats (Word, PowerPoint, Excel, images, text, ebooks, Markdown) into standardized PDFs, creating a unified processing foundation.
- **Annotation Interface** provides a web-based platform to annotate layout detection, OCR, tables, figures, and formulas outputs.

- **AI/ML Models Enhancement** supports continuous models enhancement through data download and result submission API endpoints, and automatic evaluation metrics dashboard generation (latency and accuracy).

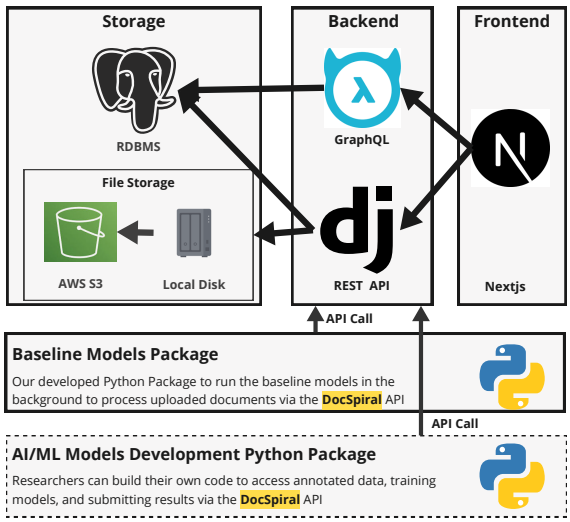

Figure 2: System Architecture Overview for **DocSpiral**

3.2 System design

The software stack of **DocSpiral** is illustrated in Figure 2, which consists of three main layers:

- **Frontend:** Built with React/Next.js[2], providing a responsive web-based user interface.
- **Backend:** Consists of two frameworks: (1) Hasura[3] to generate GraphQL endpoints, enabling rapid feature development and supporting **real-time** collaboration among annotators; (2) Django[4] for database migration management, user authentication, RESTful endpoints, and AWS S3[5] integration, etc.
- **Storage:** Uses PostgreSQL[6] for metadata, user management, annotations, evaluation metrics, and task tracking, while document files are securely stored in a private S3 bucket.

Additionally, we have developed a Python package that interacts with the platform via API calls to run baseline models and report results. Researchers can extend these functionalities by creating their own packages to download documents and annotated data, train models, perform inference using the **Progressive Models**, and submit results through the RESTful endpoints. The system is deployed in the Amazon Web Services (AWS) for scalable and secure infrastructure.

3.3 Implementation

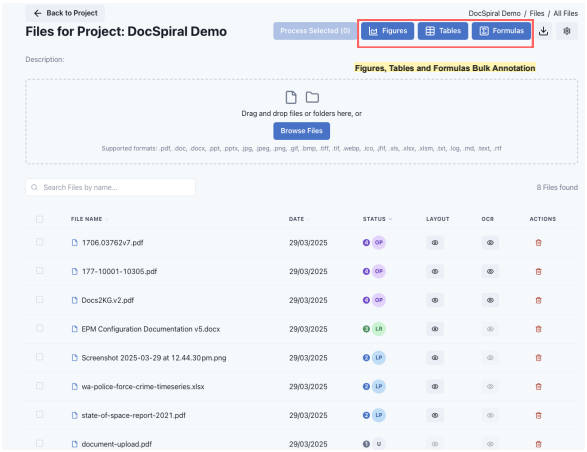

Figure 3: Documents upload and management interface, users can drag and drop allowed format documents or zipped files. Files will be uploaded to S3 bucket, and downstream tasks will be triggered.

a.) Documents Uploading: Users start by registering an account, creating a project within **DocSpiral**, and uploading documents through the interface shown in Figure 3. PDF files undergo immediate layout detection process, while other formats are first converted to PDF via **Anything2PDF**. Uploaded documents can be managed within the platform, and users can track the processing progress of each document. The system assigns the following status values: ① *Uploaded* ② *Layout detection completed* ③ *Human-reviewed layout* ④ *OCR and processing of figures, tables, and formulas completed* ⑤ *Human-reviewed model outputs from previous step*.

Once a document reaches status ② or higher, the layout *"eye"* icon becomes clickable for review. Similarly, the OCR column becomes interactive once the document reaches status ④ or above.

b.) Layout Detection Annotation: We use DocLayout-YOLO (Zhao et al., 2024d) as baseline model for layout detection. You can speficy your own baseline model when you use **DocSpiral**. This model treats each PDF page as an image and outputs bounding boxes for detected layout elements along with their corresponding labels. The data of the bounding box are represented

[2] https://nextjs.org/
[3] https://hasura.io/
[4] https://www.djangoproject.com/
[5] https://aws.amazon.com/
[6] https://www.postgresql.org/

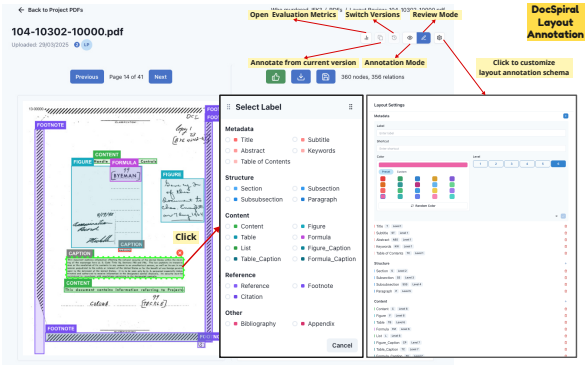

Figure 4: Layout annotation interface: user can click, add or remove bounding boxes from PDF Viewer, and assign layout labels (middle), or customize a domain specific hierarchical layout schema (right).

as $[x_{min}, y_{min}, width, height]$ in normalized coordinates. Document layouts vary significantly across domains, making it difficult to develop a universal layout detection model. For example, our baseline model supports only a limited set of labels—content, title, figure, table, formula, footnote—and struggles with complex layouts, such as PDF forms in hospital, you may want to define a label for patient name. To address this, our system enables domain-specific customization: (1) users can define their custom layout schema with hierarchies and (2) refine annotations by reclassifying, removing or adding bounding boxes with labels (Figure 4).

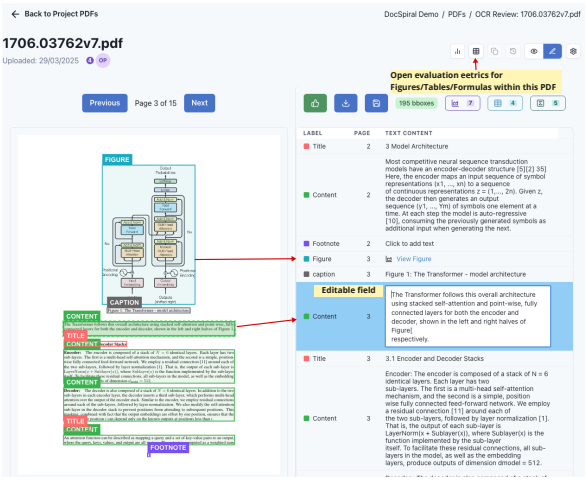

Figure 5: OCR verification and annotation interface

c.) OCR Annotation: After reviewing the layout detection results, users can save and trigger downstream processing, including the OCR process. We use PaddleOCR (PFCCLab, 2020) as the baseline OCR model for its strong performance, multilingual support, and ease of use. OCR results for each layout block appear in a table alongside their labels. The interface enables interactive navigation: clicking a bounding box in the PDF viewer scrolls the table to the corresponding row, while selecting a table row highlights the relevant section in the PDF viewer. Users can edit incorrect text directly, with changes auto-saved, as shown in Figure 5.

d.) Table, Formula and Figure Annotation: There are several models that process table images to output diverse formats for different purposes. We support multiple formats for table conversion (HTML (Smock et al., 2023), LaTeX (Xia et al., 2024), JSON (Ali Khandokar and Deshpande, 2025)) using various baselines: Pix2Text for HTML, StructEqTable for LaTeX, and a vision LLM agent for JSON extraction. For formula conversion, we output LaTeX using Pix2Text, while figure understanding leverages a vision LLM to generate descriptive text.

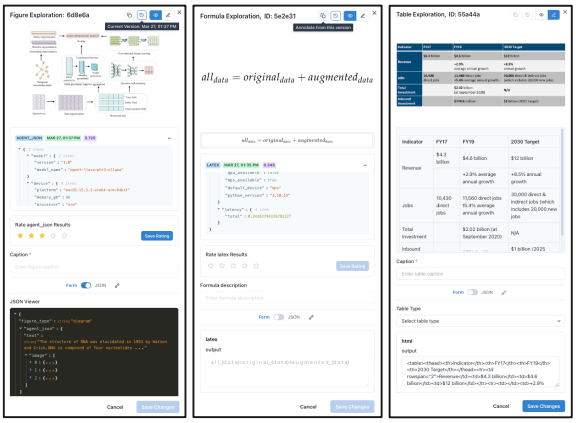

Figure 6: Figure annotation interface in `review mode` with JSON viewer (left); Formula in `review mode` showing `latex` output in form (middle); Table in `annotation mode` with editable `output` field from `html` model using schema-generated form (right).

When users click on a *Figure*, *Formula* or *Table* row, a corresponding annotation interface appears (Figure 6). Since models for different purposes require different output formats for figures, formulas, and tables, we implemented a dynamic, flexible annotation interface through **our annotation form generation feature**: users define their `Focused Model` and form schemas in settings (Figure 7), and the interface generates appropriate input fields based on the schema of the selected model. For example, selecting the `html` model for table annotation creates a TextArea field named `output`, while switching to `html_json` generates input fields for rows, caption, etc. Model outputs

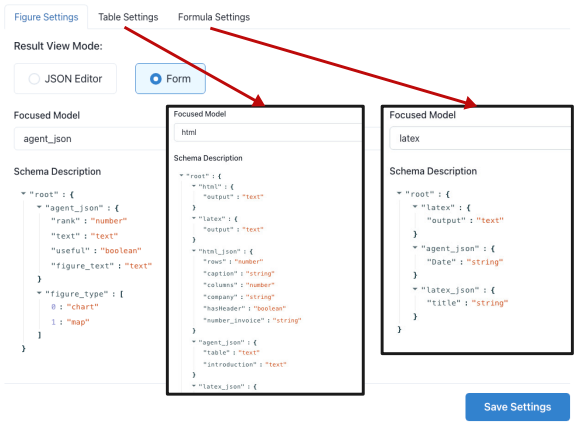

Figure 7: The settings interface allows configuring form schema and selecting a `Focused Model`. Outputs display as raw `JSON` (Figure 6 left) or as structured `Form` (Figure 6 middle/right) with fields determined by the selected model's form schema.

are prepopulated when possible to reduce manual effort. The `JSON Editor` mode allows users to examine all model outputs for better observability and to inform better annotation form schema design.

For improved efficiency, **DocSpiral** supports bulk annotation of figures, tables, and formulas across projects via buttons in Figure 3. Users can also upload standalone images for direct annotation without starting from PDF layout detection.

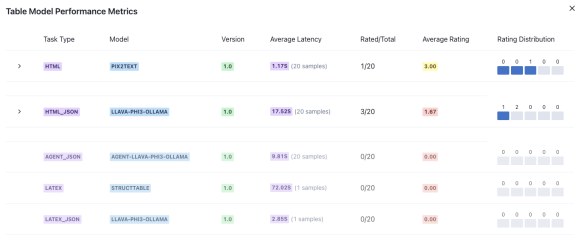

Figure 8: Table model performance dashboard displaying metrics across different output type, models, and versions. Metrics include latency, human satisfaction ratings, annotation and review progress tracking.

e.) Metrics Dashboard Generation: DocSpiral tracks objective measures (mAP for layout detection, CER/WER for OCR) and records latency for all model runs. For subjective outputs (figures, formulas, and tables processing), we implement **human satisfaction ratings** feature to quantify model-human alignment (Figure 7 left and middle). A centralized dashboard (exemplified in Figure 8) aggregates these metrics to monitor model performance and annotation progress. Other evaluation metrics can be added based on user feedback.

f.) Model Development Support: Raw data (PDFs, figures, tables, and formulas) and annotations are securely accessible via authenticated RESTful endpoints, together with submission of models outputs. Detailed instructions are available from **DocSpiral** documentation.

4 System evaluation

We quantitatively evaluated **DocSpiral**'s efficiency through an annotation experiment with 90 diverse document pages. Baseline model-assisted annotation reduced processing time from 28.4s to 16.7s per page compared to manual annotation, yielding a 41% overall time reduction. For low-quality scanned PDFs, time reduced by 75%.

Table 2: Faster-RCNN Training Performance

Metric	Initial	1st	2nd	3rd
mAP (%)	0.053	0.12	0.21	0.33

We identify three promising pathways for model spiral evolution: (1) **Traditional rule-based solutions** benefit from improved observability, enabling targeted fixes such as removing footnotes in specific locations; (2) **Deep learning models** like Faster-RCNN (Ren et al., 2016) can be fine-tuned or redesigned and trained using annotated data; (3) **Large language models (LLMs)** can be fine-tuned for better domain-specific alignment in figure, table and formula understanding. We experiment with Faster-RCNN training for layout detection over three iterative cycles, each adding 100 new pages of data, demonstrated progressive performance gains (Table 2), validating our methodology.

5 Conclusion

This paper presents **DocSpiral**, the first integrated assistive annotation platform that employs a **Human-in-the-Spiral** paradigm to extract structured information from domain-specific, image-based documents. It delivers three innovations: end-to-end annotation interfaces, a customizable hierarchical layout schema, and dynamic annotation forms for figures, formulas, and tables. Experiments show **DocSpiral** cuts annotation time by at least 41% while human feedback and model predictions iteratively reinforce each other, steadily boosting accuracy. By freely releasing the platform—and open-sourcing it once stabilized—we aim to lower barriers to AI/ML development in document-intensive fields.

References

ABBYY. 1993. Abbyy finereader pdf. Commercial document conversion and OCR software.

Iftakhar Ali Khandokar and Priya Deshpande. 2025. Computer vision-based framework for data extraction from heterogeneous financial tables: A comprehensive approach to unlocking financial insights. *IEEE Access*, 13:17706–17723.

Justin Brooks. 2019. COCO Annotator. https://github.com/jsbroks/coco-annotator/.

Explosion AI. 2023. Prodigy pdf. PDF annotation plugin for Prodigy.

Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. 2024. A survey on rag meeting llms: Towards retrieval-augmented large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '24, page 6491–6501, New York, NY, USA. Association for Computing Machinery.

Manuel Faysse, Hugues Sibille, Tony Wu, Bilel Omrani, Gautier Viaud, CELINE HUDELOT, and Pierre Colombo. 2025. Colpali: Efficient document retrieval with vision language models. In *The Thirteenth International Conference on Learning Representations*.

gipplab. 2019. Annomathtex. https://github.com/gipplab/AnnoMathTeX. Accessed: 2025-03-19.

Robert Heeg. 2023. Possibilities of unstructured data. https://shorturl.at/RaoHc. Accessed: 2025-03-28.

Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. 2022. Layoutlmv3: Pre-training for document ai with unified text and image masking. *Preprint*, arXiv:2204.08387.

Viet-Phi Huynh, Yoan Chabot, Thomas Labbé, Jixiong Liu, and Raphaël Troncy. 2022. From Heuristics to Language Models: A Journey Through the Universe of Semantic Table Interpretation with DAGOBAH. In *Semantic Web Challenge on Tabular Data to Knowledge Graph Matching (SemTab)*.

Christopher Kermorvant, Eva Bardou, Manon Blanco, and Bastien Abadie. 2024. Callico: a versatile open-source document image annotation platform. *Preprint*, arXiv:2405.01071.

Saeid Nahavandi. 2017. Trusted autonomy between humans and robots: Toward human-on-the-loop in robotics and autonomous systems. *IEEE Systems, Man, and Cybernetics Magazine*, 3(1):10–17.

Mark Neumann, Zejiang Shen, and Sam Skjonsberg. 2021. PAWLS: PDF annotation with labels and structure. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 258–264, Online. Association for Computational Linguistics.

Nick-Barney. 2025. What is unstructured data? https://www.techtarget.com/searchbusinessanalytics/definition/unstructured-data. Accessed: 2025-03-28.

Alejandro Peña, Aythami Morales, Julian Fierrez, Javier Ortega-Garcia, Iñigo Puente, Jorge Cordova, and Gonzalo Cordova. 2024. Continuous document layout analysis: Human-in-the-loop ai-based data curation, database, and evaluation in the domain of public affairs. *Information Fusion*, 108:102398.

PFCCLab. 2020. Ppocrlabel. Annotation tool for OCR tasks based on PaddleOCR.

READ-COOP SCE. 2013. Transkribus. Platform for the automated recognition, transcription and searching of historical documents.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2016. Faster r-cnn: Towards real-time object detection with region proposal networks. *Preprint*, arXiv:1506.01497.

Monica Riedler and Stefan Langer. 2024. Beyond text: Optimizing rag with multimodal inputs for industrial applications. *Preprint*, arXiv:2410.21943.

Angela Riganti, Terence R. Farrell, Margaret J. Ellis, Felicia Irimies, Colin D. Strickland, Sarah K. Martin, and Darren J. Wallace. 2015. 125years of legacy data at the geological survey of western australia: Capture and delivery. *GeoResJ*, 6:175–194. Rescuing Legacy Data for Future Science.

Zejiang Shen, Ruochen Zhang, Melissa Dell, Benjamin Charles Germain Lee, Jacob Carlson, and Weining Li. 2021. Layoutparser: A unified toolkit for deep learning based document image analysis. *arXiv preprint arXiv:2103.15348*.

Hiroyuki Shindo, Yohei Munesada, and Yuji Matsumoto. 2018. PDFAnno: a web-based linguistic annotation tool for PDF documents. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Brandon Smock, Rohith Pesala, and Robin Abraham. 2023. Aligning benchmark datasets for table structure recognition. pages 371–386.

Michael Stewart and Wei Liu. 2020. Seq2kg: An end-to-end neural model for domain agnostic knowledge graph (not text graph) construction from text. In *Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning*, volume 17, pages 748–757.

TagTog team. 2023. Tagtog. Web application. Web-based text annotation platform for machine learning and AI with project management capabilities.

Deep Search Team. 2024. Docling technical report. Technical report.

Maxim Tkachenko, Mikhail Malyuk, Andrey Holmanyuk, and Nikolai Liubimov. 2020. Label Studio: Data labeling software. Open source software available from https://github.com/HumanSignal/label-studio.

Jianqiang Wan, Sibo Song, Wenwen Yu, Yuliang Liu, Wenqing Cheng, Fei Huang, Xiang Bai, Cong Yao, and Zhibo Yang. 2024. Omniparser: A unified framework for text spotting key information extraction and table recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15641–15653.

Bin Wang, Chao Xu, Xiaomeng Zhao, Linke Ouyang, Fan Wu, Zhiyuan Zhao, Rui Xu, Kaiwen Liu, Yuan Qu, Fukai Shang, Bo Zhang, Liqun Wei, Zhihao Sui, Wei Li, Botian Shi, Yu Qiao, Dahua Lin, and Conghui He. 2024. Mineru: An open-source solution for precise document content extraction. *Preprint*, arXiv:2409.18839.

Xingjiao Wu, Luwei Xiao, Yixuan Sun, Junhang Zhang, Tianlong Ma, and Liang He. 2022. A survey of human-in-the-loop for machine learning. *Future Generation Computer Systems*, 135:364–381.

Renqiu Xia, Song Mao, Xiangchao Yan, Hongbin Zhou, Bo Zhang, Haoyang Peng, Jiahao Pi, Daocheng Fu, Wenjie Wu, Hancheng Ye, et al. 2024. Docgenome: An open large-scale scientific document benchmark for training and testing multi-modal large language models. *arXiv preprint arXiv:2406.11633*.

Renqiu Xia, Hongbin Zhou, Ziming Feng, Huanxi Liu, Boan Chen, Bo Zhang, and Junchi Yan. 2025. Latexnet: A specialized model for converting visual tables and equations to latex code. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Siyun Zhao, Yuqing Yang, Zilong Wang, Zhiyuan He, Luna K. Qiu, and Lili Qiu. 2024a. Retrieval augmented generation (rag) and beyond: A comprehensive survey on how to make your llms use external data more wisely. *Preprint*, arXiv:2409.14924.

Weichao Zhao, Hao Feng, Qi Liu, Jingqun Tang, Shu Wei, Binghong Wu, Lei Liao, Yongjie Ye, Hao Liu, Wengang Zhou, Houqiang Li, and Can Huang. 2024b. Tabpedia: Towards comprehensive visual table understanding with concept synergy. In *Advances in Neural Information Processing Systems*, volume 37, pages 7185–7212. Curran Associates, Inc.

Weichao Zhao, Hao Feng, Qi Liu, Jingqun Tang, Binghong Wu, Lei Liao, Shu Wei, Yongjie Ye, Hao Liu, Wengang Zhou, et al. 2024c. Tabpedia: Towards comprehensive visual table understanding with concept synergy. *Advances in Neural Information Processing Systems*, 37:7185–7212.

Zhiyuan Zhao, Hengrui Kang, Bin Wang, and Conghui He. 2024d. Doclayout-yolo: Enhancing document layout analysis through diverse synthetic data and global-to-local adaptive perception. *Preprint*, arXiv:2410.12628.

Yinghao Zhu, Changyu Ren, Shiyun Xie, Shukai Liu, Hangyuan Ji, Zixiang Wang, Tao Sun, Long He, Zhoujun Li, Xi Zhu, and Chengwei Pan. 2024. Realm: Rag-driven enhancement of multimodal electronic health records analysis via large language models. *Preprint*, arXiv:2402.07016.