

From Surface to Semantics: Semantic Structure Parsing for Table-Centric Document Analysis

Xuan Li, Jiali Dong*, Raymond Wong*

University of New South Wales, Sydney, Australia

Abstract. Documents are core carriers of information and knowledge, with broad applications in finance, healthcare, and scientific research. Tables, as the main medium for structured data, encapsulate key information and are among the most critical document components. Existing studies largely focus on surface-level tasks such as layout analysis, table detection, and data extraction, lacking deep semantic parsing of tables and their contextual associations. This limits advanced tasks like cross-paragraph data interpretation and context-consistent analysis. To address this, we propose DOTABLER, a table-centric semantic document parsing framework designed to uncover deep semantic links between tables and their context. DOTABLER leverages a custom dataset and domain-specific fine-tuning of pre-trained models, integrating a complete parsing pipeline to identify context segments semantically tied to tables. Built on this semantic understanding, DOTABLER implements two core functionalities: table-centric document structure parsing and domain-specific table retrieval, delivering comprehensive table-anchored semantic analysis and precise extraction of semantically relevant tables. Evaluated on nearly 4,000 pages with over 1,000 tables from real-world PDFs, DOTABLER achieves over 90% Precision and F1 scores, demonstrating superior performance in table-context semantic analysis and deep document parsing compared to advanced models such as GPT-4o.

1 Introduction

Documents are vital carriers of information across domains such as government, enterprise, and science, playing a foundational role in sectors like finance, healthcare, and academia [1–4]. As noted by UNESCO, they are essential for global knowledge transmission and cultural preservation [5]. Among document components, tables are the primary medium for structured data, often central to industrial document analysis tasks. For example, in the financial sector, analysts often need to retrieve revenue definitions along with relevant tables from reports. In the legal domain, contract reviewers must connect clauses with compensation tables. In the public sector, policy analysts frequently extract demographic summaries and related statistics from lengthy government reports. These tasks involve heterogeneous document structures, which makes the semantic table-text association essential for efficient and accurate information access.

Extensive research has explored automated analysis and extraction of information from documents and their tables [6]. As the dominant document format, PDF is widely adopted due to its lightweight

* Corresponding authors. Emails: jiali.dong@unsw.edu.au, ray.wong@unsw.edu.au

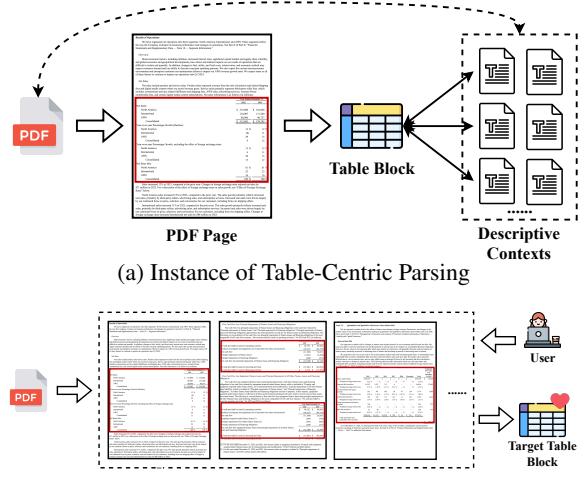


Figure 1: Example Applications of Semantic Structure Analysis

nature, cross-platform compatibility, and consistent layout [4]. However, its page-based architecture encodes text and tables as embedded graphical elements, making direct parsing difficult and significantly increasing analysis complexity [1]. Current studies mainly focus on shallow visual-level structural analysis, broadly falling into three directions. (1) document layout analysis [7, 8], which identifies regions such as text blocks, tables, and figures; (2) table detection [4, 9], which localizes tables within pages; and (3) table structure extraction and recognition [10], which reconstructs tables by parsing their structure and recognizing embedded text and data. Despite these advances, existing methods largely remain at the visual structural level, converting embedded content into machine-readable forms but lacking a deep understanding of document semantics, table content, and contextual relationships [8, 11].

For instance, as depicted in Figure 1a, for table data, merely extracting content and converting it to machine-readable formats is insufficient; analyzing contextual information is equally crucial. This context explains the data, its intended use, and underlying logical relationship foundations for semantic understanding and advanced reasoning. Besides, as shown in Figure 1b, documents often contain multiple tables, but practical analysis typically focuses on task-relevant ones. In large document collections, efficiently retrieving domain-specific tables is key to effective information use. Thus, shallow visual analysis is inadequate for complex tasks, deep semantic

parsing is indispensable for robust information extraction.

However, implementing table-centric semantic document and table parsing presents multiple challenges. First, as a page-description format, PDF embeds content as images or vector graphics without inherent structural annotations, making elements such as text and tables difficult to parse directly [6]. Currently, no comprehensive solutions exist for document-level semantic segmentation and extraction, and achieving efficient and accurate semantic partitioning in complex documents remains a significant challenge. Second, analyzing the semantic relationship between table blocks and text blocks constitutes another core challenge. Both are unstructured, lacking explicit links, which hinders direct semantic association. While natural language processing (NLP) techniques can extract implicit semantic relations, the absence of high-quality datasets modeling table-context associations in documents limits the training of traditional NLP models [12, 13]. Moreover, although large language models (LLMs) possess strong general understanding capabilities, their performance is constrained by training data, and pervasive hallucination issues further impede precise semantic relation modeling [14].

To this end, we propose DOTABLER, which to the best of our knowledge is the first framework for table-centric semantic document parsing. DOTABLER integrates multiple shallow-level document analysis modules to construct a complete preprocessing pipeline, including document segmentation, layout analysis, and optical character recognition (OCR), providing support for subsequent semantic parsing. Based on this, we developed the first semantic-level dataset modeling table-text relationships and trained the Table-Text Association Model (TTAM) as the core component. Leveraging TTAM, DOTABLER implements two key functionalities: document semantic structure parsing and domain-specific table retrieval. We evaluated DOTABLER on nearly 4,000 pages of real-world PDF documents containing over 1,000 tables. The results show that DOTABLER achieved the precision and F1 scores of over 90% in the semantic analysis of the table context, significantly outperforming advanced models such as GPT-4o, Gemini-2.0, and Claude-3.5, while delivering orders of magnitude improvements in execution efficiency. In summary, the key contributions of this study are as follows:

- We propose the first PDF semantic-level dataset modeled around table-centric structures and train the TTAM to effectively analyze relationships between tables and their contextual content.
- We design DOTABLER, which integrates a complete document preprocessing pipeline and semantic relationship analysis of PDF elements, enabling both semantic structure parsing and domain-specific table retrieval.
- We conduct a comprehensive evaluation of DOTABLER on nearly 4,000 pages of real-world PDF documents, demonstrating its superior performance and practical utility in semantic structure parsing. The source code of DOTABLER and the experiment datasets are available at <https://github.com/xuan084/DoTabler2025>.

2 Related Works

2.1 Table Extraction and Recognition

Modern Table Extraction (TE) frameworks often adapt generic object detection models, such as Faster R-CNN and Mask R-CNN, to the specific tasks of table detection and segmentation, achieving substantial performance improvements [3]. More advanced models, including Cascade Mask R-CNN [15] and Transformer-based DETR [16], have further enhanced detection precision, particularly for complex layouts. Enhancements like Deformable DETR (DDETR) im-

prove multi-scale feature representation, mitigating the convergence issues and performance limitations of standard DETR models [17]. In addition, Tc-OCR, a hybrid framework that integrates DETR, Cascade TabNet, and PP OCR v2 into a hybrid architecture to improve table extraction in scanned and noisy PDFs [18]. Building on this, retrieval-augmented OCR models trained on domain-specific datasets yield improved text recognition for tables in financial, legal, and regulatory documents [19]. However, relatively few studies have explored context-aware information extraction that incorporates both tables and their surrounding textual context.

2.2 Key Information Extraction

Key Information Extraction (KIE) from documents centers on accurately identifying and structuring semantically meaningful textual content. Existing KIE approaches can be broadly categorized into OCR-dependent and OCR-free models [6].

The OCR-dependent methods traditionally rely on sequence labeling of OCR outputs, often enhanced by layout-aware or graph-based representations that capture spatial and structural relationships between text segments [7, 8, 20]. Auxiliary detection and linking models are introduced to model complex interdependencies among text blocks [21, 22]. Recent generation-based approaches frame KIE as a structured generation problem, simplifying decoding by directly generating entities as key-value pairs, further improving adaptability across tasks [13]. In contrast, OCR-free methods aim to bypass traditional OCR pipelines entirely by incorporating text-reading capabilities directly into end-to-end architectures. Models like Donut and other sequence-to-sequence (Seq2Seq) frameworks [23, 24] are pre-trained with document image-to-text generation objectives and can directly produce structured text representations.

2.3 LLM-based Text Semantic Analysis

The advent of LLMs introduces the ability to model rich contextual embeddings, enabling a more nuanced understanding of the semantic relationships between entities [11]. Models like DocuNet and Longformer incorporate sparse attention or sliding window mechanisms, allowing for effective modeling of documents with thousands of tokens [25, 26]. The encoder-decoder architectures such as GPT-3, T5, and Flan-T5 support multirelation and multihop extraction without relying on rigid predefined relation types [27–29].

For the extraction of table content relations, LLMs have been successfully applied for the extraction of clinical information [2]. Beyond predefined table structures, DynoClass, a self-adaptive system, detects table classes dynamically without requiring predefined ontologies. This approach is particularly beneficial for evolving datasets, such as those encountered in business intelligence and market research, where table formats frequently change [30].

Despite these advances, challenges remain in integrating multimodal document signals (e.g., text + tables) and ensuring that extracted relations are coherent and factually consistent across modalities. Our work builds on this line of research by combining document understanding with specialized modules for table extraction, enabling a unified approach that captures both text-based and table-based semantic relationships.

3 Methodology

The overall workflow of DOTABLER is illustrated in Figure 2. Given a target PDF, we begin with Document Structure Preprocessing (3.1),

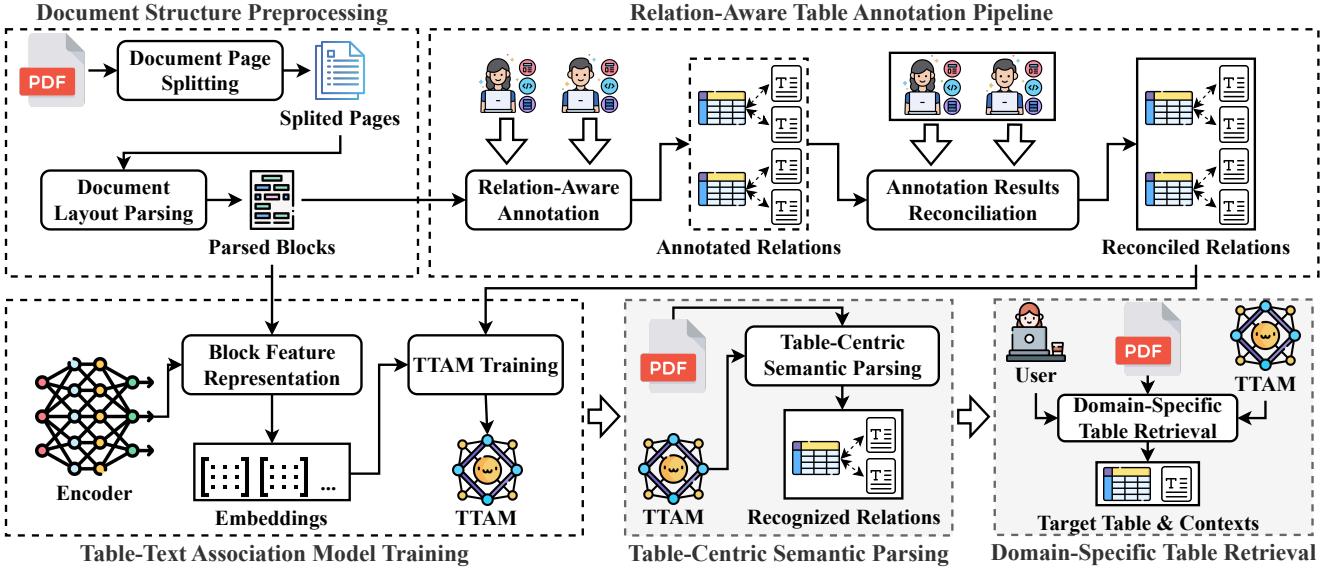


Figure 2: The overall workflow of DoTABLER

which includes page segmentation, layout element detection, and OCR-based text recognition. Next, we perform Relation-Aware Table Annotation (3.2), where paragraphs associated with each table are manually annotated. Based on the annotated data, we develop the Table-Text Association Model (TTAM), which captures semantic relationships between tables and textual content through learned semantic-level feature representations (3.3). Built upon TTAM and the structural preprocessing, DoTABLER enables two downstream capabilities: Table-Centric Semantic Parsing (3.4) and Domain-Specific Table Retrieval (3.5).

3.1 Document Structure Preprocessing

PDF files are stored in binary format and typically lack structured semantic annotations, making direct content analysis challenging [20, 31]. To overcome this, we adopt a mainstream image-based processing strategy [10], rendering each PDF page into an image to facilitate downstream structural and content analysis.

Our pipeline begins by segmenting each PDF into individual pages and converting each page into an image. We then perform document layout analysis using an object detection model fine-tuned on the PubLayNet dataset [4], which segments each page into a set of visual blocks and classifies them into semantic categories. PubLayNet follows an object detection annotation scheme, labeling blocks as one of five classes: *Text*, *List*, *Table*, *Title*, or *Figure*. We use Faster R-CNN [32] as the backbone detection model and fine-tune it on PubLayNet to enhance layout parsing accuracy. For all detected blocks classified as *Text*, *Title*, *List*, or *Table*, we apply Tesseract OCR [33] to extract the textual content. The resulting text and structural annotations serve as the foundation for modeling semantic associations between tables and relevant textual segments.

3.2 Relation-Aware Table Annotation Pipeline

Relation-Aware Annotation. We annotate the parsed blocks produced during the Document Structure Preprocessing stage, which includes page segmentation, block type classification, and OCR-based text extraction. Focusing on tables, we treat blocks labeled as *Table*

as anchors and manually identify their semantically associated textual descriptions. Since *List* blocks often contain descriptive content relevant to tables, we annotate them alongside *Text* blocks when evaluating their relationship to *Table* blocks. Throughout this paper, the term *Text block* refers collectively to both *Text* and *List* blocks.

The annotation process follows these guidelines:

- **Number Matching:** Label any text block that explicitly references a table by its number as related;
- **Semantic Supplement:** Include additional paragraphs that do not explicitly mention the table number but are semantically relevant;
- **Completeness Check:** Ensure each table block has at least one associated text block. If none, determine whether it reflects an annotation oversight or a genuine lack of textual reference.

Each table annotation within a PDF document is represented as a triplet: $\langle \text{Table-ID}, \text{Page-ID}, (\text{Related Paragraphs}) \rangle$, where Table-ID uniquely identifies a Table block, Page-ID denotes the page that the table appears, and (*Related Paragraphs*) is the set of associated *Text* and *List* blocks.

To ensure annotation quality and reliability, we engaged two researchers with over ten years of experience in document authoring and structured content analysis. They independently annotated table-text relationships to minimize potential errors and subjective bias stemming from limited domain knowledge or interpretation variance.

Annotation Results Reconciliation. To further ensure the reliability of the annotation results, we adopted an expert consensus resolution strategy [34] to reconcile discrepancies between annotators. Following the initial phase of independent annotation, the two experts collaboratively reviewed all instances with conflicting labels. Through in-depth discussion and mutual examination of their annotation rationales, they reached a consensus on each disputed case to produce a finalized, high-quality annotation set.

3.3 Table-Text Association Model Training

TTAM Model Structure. The Table-Text Association Model (TTAM) takes as input a parsed and OCR-processed *Table* block paired with a *Text* block, and outputs a binary classification indicating whether the text semantically describes the corresponding table.

Due to the inherent complexity of document semantics, characterized by multiple semantic layers, diverse information carriers, and flexible referencing styles, capturing such semantics poses significant challenges for existing shallow table recognition models and general-purpose natural language understanding systems. These models often fall short in capturing document-level semantic structures, particularly in the absence of datasets explicitly designed for this purpose. To address this gap, TTAM leverages an annotated table-text association dataset and builds upon pretrained natural language understanding models. This design enables TTAM to acquire table-centric structural and semantic knowledge, facilitating deep and context-aware semantic parsing within complex documents.

Specifically, TTAM frames the table-text relation task as a sentence-pair classification problem, where each input pair $(b_{\text{table}}, b_{\text{text}})$ consists of a table block and a text block. A pretrained model \mathcal{M} encodes this pair into a contextual representation, which is passed to a classifier \mathcal{C} to predict whether the pair is “related” or “unrelated”, as formalized in Equation (1). When combined with DOTABLER’s document preprocessing module, TTAM enables the parsing of inter-block semantic relationships, thereby facilitating document-level semantic understanding. Notably, TTAM is designed to be model-agnostic and has been successfully instantiated with various pretrained architectures, including BERT [35], BART [36], and RoBERTa [37]. Its modular design allows for the integration of other transformer-based models, offering adaptability to different analytic requirements and computational environments.

$$\hat{y} = \mathbb{I}\{\text{Softmax}(\mathcal{C}(\mathcal{M}(b_{\text{table}}, b_{\text{text}}))) \geq \theta\} \quad (1)$$

Training Strategy. During training, we construct training samples based on the annotation results obtained from the Relation-Aware Table Annotation phase and train TTAM using a cross-entropy loss function. Specifically, for each table identified by its Table-ID, we create positive samples by pairing the corresponding *Table* block with each associated *Text* block from the annotated set of (Related Paragraphs), indicating a semantic association. To construct negative samples, we randomly select an equal number of *Text* blocks from the same document that are not included in the (Related Paragraphs) and pair them with the table block to denote non-association. This sampling strategy ensures a balanced distribution of positive and negative examples for effective training.

$$\mathcal{L}_{\text{CE}}(i) = -(y_i \log p_i + (1 - y_i) \log(1 - p_i)) \quad (2)$$

Each constructed sample, consisting of a table block and a text block, is treated as a sentence pair $(b_{\text{table}}, b_{\text{text}})$ and fed into the TTAM model for relation classification. Specifically, the sentence pair is first encoded by a pretrained language model \mathcal{M} , which produces contextualized embeddings. These representations are then passed to a classifier \mathcal{C} to predict the probability p_i that the pair is the semantically “related” class. The model is optimized using the binary cross-entropy loss, as defined in Equation (2), where $y_i \in \{0, 1\}$ denotes the ground-truth label of sample i , and p_i is the predicted probability of the “related” class. This training process guides TTAM to effectively capture semantic associations between tables and text blocks, enabling robust document-level semantic parsing.

3.4 Table-Centric Semantic Parsing

Building upon TTAM and the document structure preprocessing pipeline, DOTABLER enables **Table-Centric Semantic Parsing**, with the workflow depicted in Equation (3). Given a PDF document

D , DOTABLER first applies the preprocessing pipeline $\mathcal{P}(\cdot)$, which segments D into discrete layout blocks and assigns semantic types, resulting in a block set $\mathcal{B} = \mathcal{P}(D)$. From \mathcal{B} , all blocks labeled as *Table* are extracted as $\mathcal{T} = \{b \in \mathcal{B} \mid \text{type}(b) = \text{Table}\}$. Each table block $t \in \mathcal{T}$ is treated as an anchor. For each anchor t , DOTABLER invokes TTAM to determine which *Text* or *List* blocks — collectively denoted as $\mathcal{S} = \{b \in \mathcal{B} \mid \text{type}(b) \in \{\text{Text}, \text{List}\}\}$ — that are semantically associated with t . The subset of related text blocks for table t is then computed as $\mathcal{R}_t = \{s \in \mathcal{S} \mid \text{TTAM}(t, s) = 1\}$. Through this process, DOTABLER performs fine-grained, table-centered semantic parsing, extracting each table along with its associated text, enabling comprehensive document-level semantic analysis.

$$\begin{aligned} \text{Parse}(D) &= \{(t, \mathcal{R}_t) \mid t \in \mathcal{T}, \\ \mathcal{R}_t &= \{s \in \mathcal{S} \mid \text{TTAM}(t, s) = 1\} \end{aligned} \quad (3)$$

As illustrated in Figure 1a, this application example helps to demonstrate the full process of DOTABLER performing Table-Centric Semantic Parsing. The input PDF consists of multiple pages, each containing various page elements. DOTABLER first conducts preprocessing, including page segmentation and layout analysis, to detect individual page blocks and their corresponding types — that is, it successfully identifies the table within the PDF page as a table block. It then uses each *Table* block as an anchor to identify semantically associated *Text* blocks across the document.

By analyzing the surrounding textual content, DOTABLER successfully identifies six paragraphs that describe or interpret the content of the table block, as highlighted in the figure. In the era of large-scale data, semantic-level document parsing offers a powerful approach for extracting salient information from complex, multimodal documents, substantially reducing the manual effort required for downstream analysis and decision-making.

3.5 Domain-Specific Table Retrieval

Another core capability of DOTABLER is **Domain-Specific Table Retrieval**, which accepts as input a user-defined natural-language query and a target PDF document and returns as output a set of tables — together with their associated descriptive text segments — that are semantically relevant to the query.

To enable this functionality, DOTABLER first performs Table-Centric Semantic Parsing to segment the document into a set of candidate table blocks $\{t_i\}_{i=1}^N$, each representing a distinct *Table* region extracted through document layout analysis. For semantic matching, DOTABLER adopts a fine-tuned RoBERTa cross-encoder to jointly encode the natural language query q and each candidate table t_i . Specifically, each input pair (q, t_i) is tokenized and fed into the encoder to produce a contextualized representation of the [CLS] token, which is further passed through a scoring layer to compute a scalar relevance score s_i , as formalized in Equation (4):

$$\begin{aligned} s_i &= \text{Score}(q, t_i) \\ &= \mathbf{w}^\top \cdot \text{RoBERTaCLS}(q, t_i) + b, \quad \forall i = 1, \dots, N \end{aligned} \quad (4)$$

All candidate tables are then ranked based on their scores $\{s_i\}$ in descending order, and the top- k tables are returned:

$$\mathcal{R}_{\text{top}k} = \text{TopK}\left(\{(t_i, s_i)\}_{i=1}^N\right) \quad (5)$$

As illustrated in Figure 1b, the annual report of listed companies contains multiple tables, each presenting distinct information.

DOTABLER first segments the document into structured blocks and identifies all regions classified as *Table*. Each table block is paired with the user-defined query and encoded jointly using the cross-encoder. The retrieval score is then computed via the scoring head, and top-ranked tables are returned. This retrieval mechanism is trained using a margin-based ranking loss over positive and negative query–table pairs, ensuring that relevant tables receive higher scores than irrelevant ones. Manual validation confirms that the top-ranked tables are consistently aligned with the query intent, demonstrating the effectiveness of the semantic ranking framework.

In the era of large-scale, unstructured document corpora, this table-level retrieval capability offers an efficient and scalable solution for content navigation, alleviating the cognitive and computational burden for domain experts and analysts.

4 Experiments

4.1 Implementation Details

Document Structure Preprocessing. We employ pdf2image [38] to segment the document into individual pages and export them as .jpg images. In the document layout parsing stage, we utilize Faster R-CNN to perform layout analysis on page images. The model is implemented within the Detectron2 [39] framework, using the officially released PubLayNet dataset [40].

TTAM Implementation. TTAM leverages encoder-based pre-trained models to extract feature representations from input data. Specifically, it integrates three pretrained models, BERT [35] (bert-based-uncased), BART [37] (bart-base), and RoBERTa [36] (roberta-base). The model downloading, deployment, and related operations are all implemented using the Hugging Face Transformers library [41].

Experimental Environment. All experiments, including model training and evaluation, were conducted on a Ubuntu 22.04 server equipped with an RTX 4090 GPU.

4.2 Experimental Settings

Dataset. As no publicly available dataset currently exists for document-level semantic structure analysis, particularly with a focus on table-centric semantics, we constructed, to the best of our knowledge, the first dataset explicitly designed to model document semantic structures with tables as primary anchors. This dataset was developed following the data annotation pipeline detailed in Section 3.2. Specifically, we collected documents from the following two domains:

- **arXiv** [42]: An open-access repository of scholarly papers covering the natural sciences, engineering, and related fields. Specifically, in April 2025, we retrieved the 5,000 most recently uploaded paper PDFs from arXiv and randomly selected 130 of them, excluding those that employed uncommon formatting templates, as the subjects of our study.
- **PubMed Central** [43]: An open-access database of literature in the life sciences and medical domains, offering a rich source of standardized, table-intensive documents. In April 2025, we retrieved the 5,000 most recently uploaded paper PDFs from PubMed Central and randomly selected 120 of them, again excluding those with non-standard formatting templates.

Table 1 summarizes the dataset statistics. #PDF, #Page, #Table Block, and #Text Block denote the number of source PDFs, total pages, extracted tables, and associated descriptive text blocks.

Table 1: Details of the Constructed Dataset

Source	#PDF	#Page	#Table Block	#Text Block
arXiv	125	2,408	741	1,101
PubMed Central	102	1,544	320	523
Sum	227	3,952	1,061	1,624

For TTAM Evaluation: We annotated 3,248 table-text pairs (1,624 positive, 1,624 negative), and randomly split them (7:3) into 2,273 training and 975 test samples.

For Domain-Specific Table Retrieval: From 100 sampled tables, two domain-specific queries were created per table – one from the table title and one via expert consensus – yielding 200 <query, table> pairs. After filtering incomplete or ambiguous cases, the final set includes 129 training and 53 test samples.

Baselines. As there is currently no established method in the academic literature that analyzes the semantic structure of PDF documents using table-centric cues, we employ capable LLMs as experimental baselines. Specifically, we utilize GPT-4o, Gemini-2.0 Flash, and Claude 3.5, paired with a carefully constructed prompt to form our baseline evaluation framework.

- **GPT-4o [44]:** Developed by OpenAI, GPT-4o is a state-of-the-art multimodal model supporting text, vision, and audio inputs. Its strong understanding of tables and document layouts makes it a suitable baseline for this task.
- **Gemini-2.0 Flash [45]:** Proposed by Google DeepMind, this is a highly efficient multimodal model optimized for fast, high-quality processing of text and structured visual data, making it a strong candidate for baseline comparison.
- **Claude 3.5 [46]:** A multimodal language model capable of interpreting complex document structures, including tables, and is included as a baseline to assess semantic understanding in document parsing.

To enable the LLM to analyze the relationship between table blocks and text blocks, we designed the following prompt to guide the model’s understanding of the task and fully leverage its capabilities, in which [table_content] and [text_content] denote the OCR-scan results of table blocks and text blocks, respectively:

Prompt: You are an expert in document analysis. Your task is to determine whether the provided text block is a descriptive explanation of the given table block. Please reply with only a single number:

Reply ‘1’ if the text block describes or explains the table block.

Reply ‘0’ if the text block is unrelated to the table block.

Here is the content:

- Table Block: [table_content]
- Text Block: [text_content]

Metrics. We define the following metrics to quantitatively evaluate the performance of DOTABLER:

• **Precision, Recall, and F1 of Text-Table Relation (%):** Evaluate the TTAM’s ability to correctly link text blocks to table blocks. Positive samples represent true associations, while negative samples represent unrelated pairs. Metrics are computed based on true positives (TP), false positives (FP), and false negatives (FN).

• **Document-Level Semantic Parsing Correctness:** The number of PDF documents where table-text associations are correctly recognized, covering completely correctness and partly correctness.

Table 2: Performance Evaluation of Table–Text Block Linking

Scheme	TP	FP	TN	FN	Precision	Recall	F1
GPT-4o	168	19	450	338	89.84	33.20	48.48
Gemini-2.0	373	59	410	133	86.34	73.72	79.53
Claude-3.5	316	31	438	190	91.07	62.45	74.09
BERT	426	35	434	80	92.41	84.19	88.11
BART	444	50	419	62	89.88	87.75	88.80
RoBERTa	455	50	419	51	90.10	89.92	90.01

- **Retrieval Recall@K (%)**: Measures the proportion of relevant tables correctly retrieved within the top-K results, reflecting the effectiveness of the retrieval strategy.
- **Latency (s)**: Measures the time overhead (in seconds) required for DoTABLER to complete the analysis.

Research Questions. To evaluate the performance of DoTABLER and compare it against baseline methods, we define the following research questions (**RQs**) focusing on its TTAM model and two core functionalities: Table-Centric Semantic Parsing and Domain-Specific Table Retrieval:

- **RQ1**: Can DoTABLER’s TTAM model effectively determine whether a text block describes a specific table block?
- **RQ2**: Can DoTABLER accurately perform semantic parsing of PDF documents using tables as structural cues?
- **RQ3**: Can DoTABLER reliably retrieve relevant tables and their contextual text based on user-provided natural language queries?
- **RQ4**: Does DoTABLER outperform the baselines in time efficiency and maintain low latency?

4.3 RQ1: TTAM Performance

The evaluation results of TTAM are summarized in Table 2. TTAM supports multiple pretrained models as encoders, currently including BERT, BART, and RoBERTa. Across all configurations, TTAM consistently achieves over 85% F1 score, with RoBERTa delivering the best performance achieving: Precision of 90.10%, Recall of 89.92%, and F1 score of 90.01%, demonstrating strong capability in accurately identifying table and text blocks. Error analysis reveals that TTAM’s failure cases primarily involve overly generic text descriptions that lack specific references to table elements such as headers or numerical data. For example, in document Doc-A¹, the table presents a fluctuation in mean absolute error relative to a variable. However, because the table contains minimal text (primarily numbers) and the accompanying paragraph only describes trends without citing specific values, TTAM incorrectly classifies the pair as unrelated. Notably, such cases are also difficult to resolve even through manual inspection. For comparison, we evaluated three state-of-the-art LLMs: GPT-4o, Gemini-2.0 Flash, and Claude 3.5. While these models achieve relatively high precision. For example, Claude 3.5 attains 91.07% Precision, indicating reliable identification of relevant table-text pairs – they suffer from substantial false negatives. Claude 3.5, in particular, produces 190 false negatives, resulting in a Recall of only 62.45%, reflecting significant omissions of table-associated segments.

4.4 RQ2: Document-Level Semantic Parsing

In this section, we conduct a document-level analysis of the TTAM test set to compare the performance of different methods from a

¹ The document name is anonymized in accordance with platform policies.

Table 3: Results of Table-Centric Document Semantic Parsing

Scheme	All Correct	POS Correct	NEG Correct	#Sum
GPT-4o	109	119	183	
Gemini-2.0	113	147	159	
Claude-3.5	114	137	170	
DoTABLER	128	166	155	193

Table 4: Results of Domain-Specific Table Retrieval

Scheme	Retrieval Recall@K
DoTABLER @K=1	71.70
DoTABLER @K=2	84.91
DoTABLER @K=3	88.68

table-centric perspective. Each document in the test set contains multiple table-text pairs, which may be either descriptively related or unrelated. We evaluate performance using the following three criteria: (1) the number of documents in which all table-text relationships are correctly identified (**All Correct**); (2) the number of documents in which all descriptively related table-text pairs are correctly identified, i.e., positive samples (**POS Correct**); and (3) the number of documents in which all unrelated table-text pairs are correctly identified, i.e., negative samples (**NEG Correct**). These metrics respectively assess each method’s capacity for comprehensive semantic structure analysis, accurate identification of relevant content, and avoidance of false associations. The evaluation is conducted on the TTAM test set, which includes 193 documents.

Results are shown in Table 3. Evidently, DoTABLER achieves the highest performance across All Correct and POS Correct, outperforming all three decoder-based LLMs. This demonstrates DoTABLER’s superior ability to extract semantically relevant content using tables as anchors and its stronger document-level semantic understanding compared to state-of-the-art generative models. It is important to note that although LLMs tend to adopt conservative decision strategies, they often produce fewer false positives but more false negatives. As a result, they show relatively better performance on the NEG Correct metric in this limited test set. However, since the primary goal of semantic parsing is to accurately identify related table–text associations, the modest performance of LLM-based approaches in this area highlights their limitations.

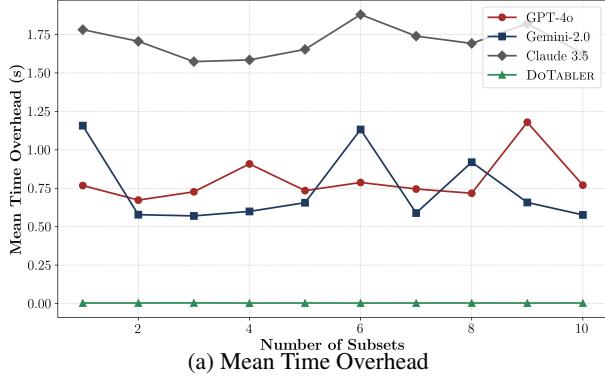
4.5 RQ3: Domain-Specific Table Retrieval

Table 4 reports the results of the domain-specific table retrieval evaluation. Given a natural language query, DoTABLER employs a TTAM-based ranking strategy to compute the semantic relevance between the query and all tables within a PDF document, and returns the top- K ranked tables as retrieval results. When $K = 1$ – i.e., retrieving only the most relevant table—the retrieval recall (Recall@1) reaches 71.70%. As K increases to 3, the recall improves to 88.48%.

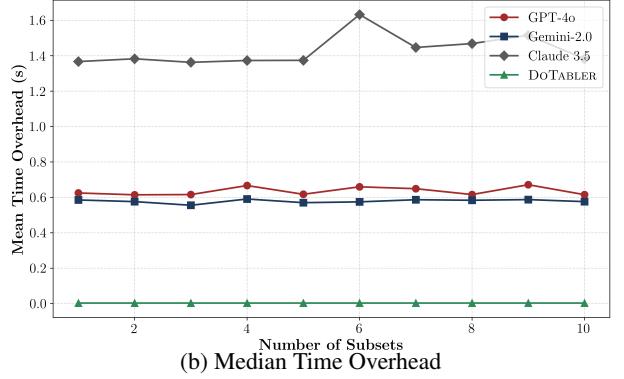
It is worth noting that PDF documents often contain multiple structurally diverse tables; in some extreme cases, a single document may include a large number of tables. For instance, one test case contains 27 tables. Accurately retrieving the query-relevant table under such conditions poses a significant challenge. Nonetheless, DoTABLER demonstrates strong robustness and practical effectiveness across these complex scenarios.

4.6 RQ4: Efficiency Evaluation

In this section, we evaluate the time overhead of DoTABLER on document semantic parsing and compare it with LLM-based approaches. Specifically, we measure both the average and median time overhead



(a) Mean Time Overhead



(b) Median Time Overhead

Figure 3: Time Overhead of Distinct Schemes regarding Subsets

Table 5: Overall Time Cost of Distinct Schemes

Scheme	Mean Time Cost (s)	Median Time Cost (s)
GPT-4o	0.8008	0.6201
Gemini-2.0	0.7434	0.5763
Claude-3.5	1.7054	1.4214
DoTABLER	0.0035	0.0031

for each method using the 975 test pairs from RQ2 (semantic parsing). To mitigate the potential impact of data distribution bias, we randomly divide all test samples into 10 batches evenly and compute the mean and median time overhead within each group.

The results are presented in Table 5, Figure 3a and Figure 3b. DoTABLER achieves both average and median time overheads below 0.01 seconds, demonstrating exceptional efficiency. This is largely attributed to TTAM, an encoder-based, moderately sized pre-trained model that runs locally during inference, allowing fast execution. In contrast, the three LLM-based baselines show significantly higher latency, with overheads approximately two orders of magnitude greater. For instance, Claude 3.5 exhibits average and median time overheads exceeding 1 second. In the batch-wise analysis shown in Figures 3a and 3b, DoTABLER consistently outperforms all LLMs across both metrics, maintaining a significant time advantage. This result highlights the high efficiency of DoTABLER, which is particularly critical in real-world scenarios involving large volumes of documents with dense table-text structures.

5 Discussion

5.1 Model Effectiveness and Comparative Analysis

Through a series of comprehensive evaluations, we demonstrated the effectiveness of DoTABLER in capturing semantic associations between tables and related textual segments, highlighted the strong performance of DoTABLER in table-centric semantic parsing tasks.

While advanced generative language models such as GPT-4o and Gemini exhibit impressive reasoning and multimodal capabilities, DoTABLER – powered by RoBERTa – outperforms them in structured document understanding. Trained with masked language modeling, RoBERTa is well-suited for capturing fine-grained contextual dependencies and ensuring precise alignment between structured components. Its bidirectional encoder architecture and lower susceptibility to hallucinations allow it to excel in tasks like associating tables with relevant paragraphs.

In contrast, decoder-only models such as GPT-4o and Gemini are optimized for fluent text generation, which makes them more prone to hallucination – producing outputs that are semantically plausible but factually inaccurate or unsupported [14]. This weakness

poses challenges in tasks requiring high-precision, cross-structural reasoning. The superior performance of RoBERTa’s discriminative approach has also been corroborated by other recent studies [47–49], further validating its effectiveness in structured document analysis.

5.2 Limitations and Future Works

Despite its overall effectiveness, DoTABLER has certain limitations. First, it depends on existing preprocessing tools such as document layout analysis and OCR. While these techniques are generally reliable, they still struggle with complex layouts, non-standard templates, and scanned documents containing embedded tables. These challenges can affect parsing accuracy. Nonetheless, DoTABLER remains effective on the majority of documents evaluated. Second, the performance of DoTABLER may degrade on low-quality documents, especially when the relationship between tables and text is vague or implicit. In rare cases, contextual descriptions refer to general trends without explicitly mentioning table headers or values, making semantic association difficult. Future efforts may focus on improving robustness to irregular document structures and enhancing the model’s ability to infer implicit semantic links.

5.3 Ethical Considerations

All data were sourced from publicly available arXiv and PubMed Central (Open Access Subset) documents, using only metadata and annotations in compliance with open-access licenses (e.g., CC-BY). No sensitive or personal information was included, and all data were used solely for academic research.

6 Conclusion

In this paper, we proposed DoTABLER, a table-centric semantic document parsing framework that integrates multiple shallow-level document analysis modules, including document segmentation, layout analysis, and OCR. This is implemented through a three-stage pipeline centered around the TTAM. To evaluate the effectiveness of our approach, we constructed a dataset comprising nearly 4,000 pages of real-world PDF documents containing 1,000 tables. Experimental results show that DoTABLER achieves highly competitive performance, even when compared with advanced LLMs. As a general-purpose framework for table-centric semantic document parsing, DoTABLER has demonstrated strong capability in extracting tables and their associated textual context. In future work, we aim to enhance DoTABLER’s capabilities and broaden its applicability across diverse domains and real-world deployment scenarios.

References

- [1] H. Li, H. Gao, and C. Wu, “Extracting financial data from unstructured sources: Leveraging large language models,” *Journal of Information Systems*, 2024.
- [2] D. Hein, A. Christie, M. Holcomb, B. Xie, A. Jain, J. Vento, N. Rakheja, A. Hamza Shakur, S. Christley, L. G. Cowell, *et al.*, “Prompts to table: Specification and iterative refinement for clinical information extraction with large language models,” *medRxiv*, 2025.
- [3] M. Li, L. Cui, S. Huang, F. Wei, M. Zhou, and Z. Li, “Tablebank: Table benchmark for image-based table detection and recognition,” in *Twelfth Language Resources and Evaluation Conference (LREC)*, 2020.
- [4] X. Zhong, J. Tang, and A. J. Yepes, “Publaynet: largest dataset ever for document layout analysis,” in *15th International conference on document analysis and recognition (ICDAR)*, 2019.
- [5] “Unesco: Memory of the world,” 2025.
- [6] J. Wan, S. Song, W. Yu, Y. Liu, W. Cheng, F. Huang, X. Bai, C. Yao, and Z. Yang, “Omniparser: A unified framework for text spotting key information extraction and table recognition,” in *Proceedings of the 42nd IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.
- [7] Z. Gu, C. Meng, K. Wang, J. Lan, W. Wang, M. Gu, and L. Zhang, “Xylayoutlm: Towards layout-aware multimodal networks for visually-rich document understanding,” in *Proceedings of the 40th IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.
- [8] Y. Huang, T. Lv, L. Cui, Y. Lu, and F. Wei, “Layoutlmv3: Pre-training for document ai with unified text and image masking,” in *Proceedings of the 30th ACM International Conference on Multimedia (MM)*, 2022.
- [9] W. Zhao, H. Feng, Q. Liu, J. Tang, B. Wu, L. Liao, S. Wei, Y. Ye, H. Liu, W. Zhou, *et al.*, “Tabpedia: Towards comprehensive visual table understanding with concept synergy,” *38th Advances in Neural Information Processing Systems (NeurIPS)*, 2024.
- [10] A. Nassar, N. Livathinos, M. Lysak, and P. Staar, “Tableformer: Table structure understanding with transformers,” in *Proceedings of the 40th IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.
- [11] Z. Zhang, B. Yu, X. Shu, T. Liu, H. Tang, W. Yubin, and L. Guo, “Document-level relation extraction with dual-tier heterogeneous graph,” in *28th International Conference on Computational Linguistics (COLING)*, 2020.
- [12] Y. Ma, Y. Zang, L. Chen, M. Chen, Y. Jiao, X. Li, X. Lu, Z. Liu, Y. Ma, X. Dong, *et al.*, “Mmlongbench-doc: Benchmarking long-context document understanding with visualizations,” in *38th Advances in Neural Information Processing Systems (NeurIPS)*, 2024.
- [13] Z. Tang, Z. Yang, G. Wang, Y. Fang, Y. Liu, C. Zhu, M. Zeng, C. Zhang, and M. Bansal, “Unifying vision, text, and layout for universal document processing,” in *Proceedings of the 41st IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [14] Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. J. Bang, A. Madotto, and P. Fung, “Survey of hallucination in natural language generation,” *ACM Computing Surveys*, 2023.
- [15] D. Prasad, A. Gadpal, K. Kapadni, M. Visave, and K. Sultanipure, “Cascadetabnet: An approach for end to end table detection and structure recognition from image-based documents,” in *CVPR Workshops*, 2020.
- [16] B. Smock, R. Pesala, and R. Abraham, “Pubtables-1m: Towards comprehensive table extraction from unstructured documents,” in *Proceedings of the 40th IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.
- [17] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, “Deformable detr: Deformable transformers for end-to-end object detection,” *arXiv preprint arXiv:2010.04159*, 2020.
- [18] A. Anand, R. Jaiswal, P. Bhuyan, M. Gupta, S. Bangar, M. M. Imam, R. R. Shah, and S. Satoh, “Tc-ocr: Tablecraft ocr for efficient detection & recognition of table structure & content,” in *Proceedings of the 1st International Workshop on Deep Multimodal Learning for Information Retrieval*, 2023.
- [19] S. Saleh, “Enhancing arabic retrieval augmented generation through language processing,” *Available at SSRN 5134287*.
- [20] S. Appalaraju, B. Jasani, B. U. Kota, Y. Xie, and R. Manmatha, “Docformer: End-to-end transformer for document understanding,” in *Proceedings of the 39th IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [21] Z. Yang, R. Long, P. Wang, S. Song, H. Zhong, W. Cheng, X. Bai, and C. Yao, “Modeling entities as semantic points for visual information extraction in the wild,” in *Proceedings of the 41st IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [22] Y. Yu, Y. Li, C. Zhang, X. Zhang, Z. Guo, X. Qin, K. Yao, J. Han, E. Ding, and J. Wang, “Structextv2: Masked visual-textual prediction for document image pre-training,” *arXiv preprint arXiv:2303.00289*, 2023.
- [23] H. Cao, C. Bao, C. Liu, H. Chen, K. Yin, H. Liu, Y. Liu, D. Jiang, and X. Sun, “Attention where it matters: Rethinking visual document understanding with selective region concentration,” in *Proceedings of the 19th IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.
- [24] G. Kim, T. Hong, M. Yim, J. Nam, J. Park, J. Yim, W. Hwang, S. Yun, D. Han, and S. Park, “Ocr-free document understanding transformer,” in *17th European Conference on Computer Vision (ECCV)*, 2022.
- [25] N. Zhang, X. Chen, X. Xie, S. Deng, C. Tan, M. Chen, F. Huang, L. Si, and H. Chen, “Document-level relation extraction as semantic segmentation,” in *Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI)*, 2021.
- [26] I. Beltagy, M. E. Peters, and A. Cohan, “Longformer: The long-document transformer,” *arXiv preprint arXiv:2004.05150*, 2020.
- [27] A. Mastropaoletti, S. Scalabrino, N. Cooper, D. N. Palacio, D. Poshyvanyk, R. Oliveto, and G. Bavota, “Studying the usage of text-to-text transfer transformer to support code-related tasks,” in *IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*, pp. 336–347, IEEE, 2021.
- [28] Z. Wan, F. Cheng, Z. Mao, Q. Liu, H. Song, J. Li, and S. Kurohashi, “Gpt-re: In-context learning for relation extraction using large language models,” in *Proceedings of the 28th Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023.
- [29] S. Wadhwa, S. Amir, and B. C. Wallace, “Revisiting relation extraction in the era of large language models,” in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, 2023.
- [30] H. Wang, E. Wu, K. Liu, and J. Liu, “Dynoclass: A dynamic table-class detection system without the need for predefined ontologies,” in *NeurIPS 2024 Third Table Representation Learning Workshop*.
- [31] Y. Huang, N. Lu, D. Chen, Y. Li, Z. Xie, S. Zhu, L. Gao, and W. Peng, “Improving table structure recognition with visual-alignment sequential coordinate modeling,” in *Proceedings of the 41st IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [32] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” *29th Advances in Neural Information Processing Systems (NeurIPS)*, 2015.
- [33] “Tesseract ocr.” <https://github.com/tesseract-ocr/tesseract>, 2025.
- [34] J. Pustejovsky and A. Stubbs, *Natural Language Annotation for Machine Learning: A guide to corpus-building for applications*. 2012.
- [35] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” in *Proceedings of the 14th Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2019.
- [36] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, “Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension,” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020.
- [37] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, “Roberta: A robustly optimized bert pretraining approach,” *arXiv preprint arXiv:1907.11692*, 2019.
- [38] “pdf2image.” <https://github.com/Belval/pdf2image>, 2025.
- [39] “Detectron2.” <https://github.com/facebookresearch/detectron2>, 2025.
- [40] “Publaynet dataset.” <https://github.com/ibm-aur-nlp/PubLayNet>, 2025.
- [41] “Hugging face transformers.” <https://huggingface.co/>, 2025.
- [42] “arxiv.org e-print archive.” <https://arxiv.org/>, 2025.
- [43] “Pubmed central.” <https://pmc.ncbi.nlm.nih.gov/>, 2025.
- [44] “Model - openai api.” <https://platform.openai.com/docs/models>, 2025.
- [45] “Gemini models | gemini api | google ai for developers.” <https://ai.google.dev/gemini-api/docs/models>, 2025.
- [46] “Anthropic - all models overview.” <https://docs.anthropic.com/en/docs/about-claude/models/all-models>, 2025.
- [47] N. K. Benkler, S. Friedman, S. Schmer-Galunder, D. M. Mosaphir, R. P. Goldman, R. Wheelock, V. Sarathy, P. Kantharaju, and M. D. McLure, “Recognizing value resonance with resonance-tuned roberta task definition, experimental validation, and robust modeling,” in *Proceedings of the Joint International Conference on Computational Linguistics, Language Resources and Evaluation (COLING)*, pp. 13688–13698, 2024.
- [48] G. RoccaBruna, M. Rizzoli, and G. Riccardi, “Will llms replace the encoder-only models in temporal relation classification?,” *arXiv preprint arXiv:2410.10476*, 2024.
- [49] Z. Cheng, L. Zhou, F. Jiang, B. Wang, and H. Li, “Beyond binary: Towards fine-grained llm-generated text detection via role recognition and involvement measurement,” in *Proceedings of the 32nd ACM on Web Conference (WWW)*, 2025.