Synthesizing Realistic Data for Table Recognition

Qiyu Hou¹, Jun Wang¹(\square), Meixuan Qiao², and Lujun Tian¹

¹ iWudao Tech {houqy,jwang,tianlj}@iwudao.tech
² Huazhong University of Science and Technology qiaomeixuan@hust.edu.cn

Abstract. To overcome the limitations and challenges of current automatic table data annotation methods and random table data synthesis approaches, we propose a novel method for synthesizing annotation data specifically designed for table recognition. This method utilizes the structure and content of existing complex tables, facilitating the efficient creation of tables that closely replicate the authentic styles found in the target domain. By leveraging the actual structure and content of tables from Chinese financial announcements, we have developed the first extensive table annotation dataset in this domain. We used this dataset to train several recent deep learning-based end-to-end table recognition models. Additionally, we have established the inaugural benchmark for real-world complex tables in the Chinese financial announcement domain, using it to assess the performance of models trained on our synthetic data, thereby effectively validating our method's practicality and effectiveness. Furthermore, we applied our synthesis method to augment the FinTabNet dataset, extracted from English financial announcements, by increasing the proportion of tables with multiple spanning cells to introduce greater complexity. Our experiments show that models trained on this augmented dataset achieve comprehensive improvements in performance, especially in the recognition of tables with multiple spanning cells.

Keywords: Table Data Synthesis \cdot Data Augmentation \cdot Table Recognition

1 Introduction

Tables, serving as a vital carrier of data, are prevalent across a wide range of digital documents. They efficiently store and display data in a compact and lucid format, encapsulating an immense volume of valuable information. However, recognizing the structures of tables within digital documents, such as PDF and images, and subsequently extracting structured data, present significant challenges due to the complexity and diversity of their structure and style [1]. In recent years, with the advancement of deep learning, new methodologies have surfaced, leading to significant progress in table recognition [21,19,18,24,16]. Deep

learning-based methods are adept at managing complex table structures and diverse styles more effectively. However, their reliance on large-scale, high-quality annotated table datasets for model training is pronounced [16,1]. The scarcity of comprehensive and intricately detailed, publicly accessible datasets emerges as a substantial barrier, impeding the further advancement of table structure recognition.

To create large-scale datasets for table recognition, some researchers have initially started by utilizing specific repositories of scientific papers or financial reports. Each document in these repositories contains tables that correspond to some form of structured source codes (such as LaTeX, XML, HTML). They facilitate large-scale annotation of table recognition data by automatically mapping the displayed tables to their corresponding structured source codes. The existing large-scale, real-world table annotation datasets [20,27,26,12,2,3] are few but have all been constructed using similar methodologies. However, the applicability of these methods for creating table annotation data is notably limited, as only a select number of fields have access to document repositories where structured source codes correspond to rendered tables. Given the substantial variations in table structures and styles across different domains and languages, the table styles featured in these datasets tend to exhibit similarities. This similarity poses challenges when attempting to apply these datasets to a broader range of domains [16,1]. These automatically annotated datasets frequently contain a significant number of annotation errors. For instance, in our sampling of over 10,000 tables from the FinTabNet dataset, we found that approximately 9% had obvious annotation errors. Furthermore, the annotation information required varies among different table recognition methods, and some of these table datasets do not completely fulfill the annotation requirements of several prevalent table recognition models [20].

To adapt more efficiently to a wider range of application domains, some researchers are exploring methods for synthesizing annotated data for table recognition. These methods largely depend on predefined structural templates and randomly selected text to generate table structures and fill in content. Moreover, they employ a web browser engine to render the synthesized tables and add annotations, following predefined style templates. However, tables rendered with HTML and CSS offer limited visual fidelity and often fall short of replicating the complex or unique appearances of tables in documents like PDFs. Consequently, they lack the richness and complexity required to accurately simulate the intricate table structures encountered in real-world scenarios.

After careful analysis of real-world tables, we discovered that generating appropriate templates for tables with complex structures, such as those found in the financial sector, is not easily achieved through random methods. The structures produced randomly often differ significantly from real complex tables, and the text randomly inserted into these tables usually deviates greatly from the context and semantics of actual tables. We have also observed that in specific application domains, such as finance, the primary challenge in table recognition stems from the fact that many tables, despite having similar structures and con-

tent, display significantly varied presentation styles. To address this issue, we propose a method that utilizes the structure and content of existing complex tables to generate high-quality, realistic synthetic datasets tailored to the target application domain. Firstly, in many scenarios, we can relatively easily access the structure and content of numerous existing tables. Taking the financial domain as an example, tables in U.S. financial disclosures are available in HTML format, allowing direct access to the table structure and the text content within the cells. Although Chinese financial announcements in PDF format do not have corresponding HTML files, they contain a large number of bordered tables. Using PDF parsing tools to extract and analyze the lines within these bordered tables, it is relatively easy to deduce the table structure and the text content in each cell. Secondly, reflecting the actual distribution of tables in the target application domain, tables can be categorized and summarized. This involves extracting and documenting the various attributes associated with the presentation styles of each table within every category. These attributes are then stored in a profile and incorporated into the candidate style set for that category. For a source table whose structure and content have been obtained, its corresponding table category can first be identified based on its content. From this category's candidate set, a profile is selected, and a small degree of randomness is introduced to the attributes of the selected profile to set the profile for the target table. This results in the transformation into a new table that is completely different in style from the original, yet still possesses a very realistic appearance. For example, Fig. 1 shows the transformation of an original bordered table into two borderless tables and one bordered table with different style, all of which have the same structure and content. Additionally, our method, based on image rendering techniques for synthesizing tables, is not constrained by the limitations of browser rendering. It can draw or paste real table appearance components from various documents, enabling the creation of more realistic complex table styles.

This paper makes the following contributions:

• A novel method for synthesizing annotation data specifically designed for table recognition has been proposed, which utilizes the structure and content of existing complex tables. This approach enables the straightforward synthesis of tables that closely resemble the actual table styles prevalent in the target domain, accompanied by comprehensive and complete annotation data.

• Utilizing the actual structure and content of tables from Chinese financial announcements, we synthesized the first large-scale table annotation dataset in the domain of Chinese financial announcements. On this basis, we trained several recent deep learning-based end-to-end table recognition models. Furthermore, we created the first benchmark for real-world complex tables in the Chinese financial announcement domain, which was used to evaluate the performance of models trained on synthetic data, thereby validating the practicality and effectiveness of the method proposed in this paper.

•Additionally, the method proposed in this paper was used to augment the table dataset extracted from English financial announcements, known as FinTab-Net, by synthesizing and increasing the proportion of more complex tables with



Fig. 1. The transformation of an original bordered table into two borderless tables and one bordered table with different style

multiple spanning cells. Experiments demonstrate that table recognition models trained on this enhanced dataset exhibit comprehensive improvements in performance, particularly in recognizing more complex tables that contain multiple spanning cells.

2 Related Work

2.1 Large-Scale Table Recognition Datasets

The current large-scale table recognition datasets are automatically annotated by utilizing a few document repositories, where the tables displayed in the documents can be associated with corresponding structured source codes.

TABLE2LATEX-450K [2], TableBank [12], and TabLeX [3] are datasets derived from articles in the Arxiv repository, where tables in PDF documents correspond to LaTeX source codes. Additionally, TableBank [12] has crawled some Word documents from the internet, linking tables in these documents to Office XML codes. PubTabNet [27] and PubTables-1M [20] were sourced from scientific papers in the PubMed Central Open Access (PMCOA) database, with tables in the PDF files paired with corresponding XML codes. Meanwhile, FinTabNet [26] compiled annual reports from S&P 500 companies, featuring tables in the PDF documents that can be associated with corresponding HTML codes. The datasets mentioned above are confined to scientific papers or financial reports, and models trained on them may not perform well in other domains. As mentioned before, these automatically annotated datasets often contain a considerable number of annotation errors. These issues encompass a variety of problems, including inaccuracies in localizing table regions, omissions of table content in annotations, structural errors in annotating headers or table content, and inconsistencies in annotations for identical structures or content within tables, among others. Except for PubTables-1M, the annotations in the other datasets do not fully cover the annotation requirements needed by several common table recognition models [20]. Specifically, both PubTabNet and FinTabNet lack annotations for the coordinates of cell bounding boxes [27,26]. Additionally, TableBank and TabLeX provide only the overarching structure of the table, without annotations for the content and coordinates of text blocks within each cell [12,3].

2.2 Randomly Synthesized Data for Table Recognition

Another series of efforts to address the scarcity of large-scale table recognition datasets involves constructing table annotation datasets through the synthesis of HTML tables.

Qasim et al. [17] initially employed four types of table templates to create a synthetic table dataset based on HTML. TableGeneration [28] further expanded the method, maintaining support for four table templates while introducing a broader range of configurable parameters, which include cell type, the number of rows and columns in a table, the quantity of merged cells, and the provision for colored cells. WikiTableSet [14] constructed a Wikipedia table extractor to harvest tables (in HTML code format) from the Wikipedia dump, and then normalized these HTML tables to align with the PubTabNet format, which involved separating table headers from data and stripping CSS and style tags. SynthTabNet [16] took a parameter-driven approach to initially generate the table structure, detailing the total number of rows and columns, header row count, types of spans (including header-only, row-only, column-only, and both row and column spans), maximum span size, and the proportion of table area covered by spans. Appropriate content templates are then selected and combined with purely random text to produce synthetic content. A collection of styling templates is manually curated, and a style is randomly selected to determine the appearance of the synthesized table. In a similar vein, the ComplexTable dataset [1] is synthetically generated using an auto HTML table creator, which produces table images alongside corresponding structured HTML code. Notably, ComplexTable features a significantly higher frequency of complex tables compared to SynthTabNet, and it showcases a more diverse range of table styles within the dataset.

The aforementioned methods employ HTML for table synthesis and CSS to define table styles, followed by the use of a Web Browser engine to render the table images. However, this approach has its limitations. Often, it falls short in replicating the complex or distinctive visual effects that tables in documents, such as PDFs, typically present.

2.3 Table Augmentation

Data augmentation is also a common method for acquiring table data. TabAug [11] introduced a novel data augmentation technique that primarily relies on two fundamental operations: Replication and Deletion, applied to rows and columns. Umer et al. [22] developed augmentation techniques that encompass clustering, fusion, and patching of table images. Ichikawa [10] introduced novel labelinvariant table augmentation techniques focused on the edge-based region, demonstrating their substantial impact, particularly when training with limited datasets. Liu et al. [13] enhanced existing datasets by employing two types of image distortion algorithms, aiming to simulate distractors introduced by the capture device.

This paper is mainly concerned with the acquisition of large-scale table annotation data encompassing diverse styles. The data augmentation methods mentioned do not significantly alter table styles, hence they differ somewhat from the core issue addressed in this paper.

3 Our Approach for Synthesizing Annotated Tables

If we aim to transform tables from documents into structured data for storage in databases or knowledge graphs within a specific domain, enabling further indepth analysis and applications, it becomes essential to parse the structure of these tables to grasp their semantics. In many application domains, although the content of some tables across different documents may be very similar, the styles of these tables with similar content often exhibit significant differences. Fig. 2 shows six real examples of tables from the Financial Statements, specifically the "Accounts Receivable - Disclosed by Provision Method" category. These tables, belonging to the same category, have similar content but vastly different visual styles. Inspired by this observation, we propose synthesizing new target tables by leveraging the structure and content of existing complex tables, while applying completely different yet realistically plausible styles to these target tables. This approach ensures that the synthesized complex tables more closely resemble realworld scenarios than those produced by methods relying on randomly generated structures and content.

Fig. 3 illustrates the overall structure of the table annotation data synthesis method proposed in this paper, which involves extensive handling of text and lines within tables. For tables generated via PDF coding, PDF parsing tools, such as pdfplumber, offer a direct and efficient means to extract texts and lines. In contrast, for scanned tables, OCR can be used to detect and recognize text, and conventional methods such as the Hough Transform [4,15] or LSD [5], as well as deep learning-based line detection techniques [7,29,25,8,6,23], can be employed to identify lines within the tables.

3.1 Analysis of the Real Table Distribution in the Target Domain

To synthesize table annotation data that closely match the target application domain, a detailed investigation and analysis of the real table data's distribution

					1	用末余额												期末:	余额			
本別			账面余	钡		坏账准备		陵	面价值		类别			账面余额			坏账	准备		ACCUTA 181		
			金額	比	例	金額	计提比	例						3	金額	比例	(%)	金	90	计提比例(%	6)	PECTED 107 101
兼運计提基)	其他应			(6)		(%)	-		_	单项计机	到期信用损失的	应收账款	11,	611,586.9	1	1.34				11	,611,586.94
收款	X ILLON	(51,229,108.1	5 1	6.37 0	1,229,108.16	100.9	20			其中: 有	在担保的应收账	款	11,	611,586.9	1	1.34				- 11	,611,586.94
按组合计提坏账准备f 应应当	的其他	31	2,778,437.6) 8	3.63 15	3,122,419.55	48.	6 159	,656,01	3.05	按组合计	提坏账准备的应	收账款	854,	501,755.3)	98.66	30,336,	300.56	3	55 824	,265,454.74
其中:账龄组合		31	2,778,437.6) 8	3.63 15	3,122,419.55	48.	96 159	,656,01	3.05	其中: 购	:畚组合		854,	501,755.3)	98.66	30,336,	300.56	3	55 824	,265,454.74
合计		37	74,007,545.7	5 10	0.00 21	4,351,527.71	57.	31 159	,656,01	3.05		合计		866,	213,342.2	1	100.00	30,336,	300.56	3	50 835	,877,041.68
						荆末余顿				_												
类别				收面余 章	1	坏赂	准备		ale care a						2.07		2022年	=12月3	1日			
				金額	比例(%)	全等	(预期	作用損 を寒(%)	床留1	17.010				账面分	教	11.00		坏则	准备	t ter i t. bet	<u> </u>	长面价值
按单项计模坏账准备的	应收账表	k								_				金額	Į	比例		账囬亲	:観 1	十提比例		
按组合计模坏账准备的	应收账表	k	36,214	536.46	100.00	1,810,278.4	13	5.00	34,404,2	58.03	单项计	提坏账准备	8 096	651 17	7 90	71%	(7.64	1.031.9	93)	94 37%	45	5 619 24
其中:											按信用	风险特征组	010001				(, 10,	1,0011	,,,,	0 1107 70		
应收内部关联方			96	000.00	0.27		-	-	96,0	00.00	合计	提坏账准备	828,	777.06	5 9	.29%	(6	5,143.	71 <u>)</u>	7.86%	76	3,633.35
应收其他客户			35,118	,536.46	99.73	1,810,278.	13	5.01	34,308,2	58.03			8,925,	428.23	3		(7,70	6,175.6	64)		1,21	9,252.59
合计			36,214	536.46	100.00	1,810,278.4	13	5.00	34,404,2	58.03												
																					Barn	
			期末余額					期初余额						_		202241	2/1310	1.00.00		2021+1	27131 H	100
#N	写出企题	-	可解准备		R.H.	東武会戦	10.04	坏鬼准备	17.85	账册					度国本	NA 片以和	40 gi	₩		ELEI) 水和 占总額	-44	E/E 倍 计提
4	全朝	00	全朝	可能电 例(60)	行員	全部	00	全朝	出例 (59)	1112					金額	比例	金額	比例	金粉	1 比別	金額	比例
核单项计提标编建备 4,13	78, 274, 34	10.09	4, 178, 274, 34	108.00	-	4, 178, 274, 34	9.51	4, 178, 274, 34	103, 60		- 1											
其中: 車項金額運大許美項計量												单项金额重大并单	独计提坏账准 (备	874	14%	(874)	100%	874	1 18%	(874)	100%
环胞准备 7-3 单项金积不重大但单项计	294,276.55	5.54	2,296,136.55	100.00	-	1,394,176.55	5.23	2,298,138,55	100.00			按组合计提环账准 约次速算数 环	音 计加速在相应		4 774	740	_	_	3.20			
最优度推高 17 放在会计和区面推高 27.2	, and and a second	*34	4, 672, 384, 23	12.52	37 620 945 12	20 737 747 20	30.41	3,850,527,80	3.63	35 887 154		1X.只用异根。即 其故	11869(从外立		687	10%	-	-	5,20	3 12%	_	-
其中,												单项金额不重大但	单独计提环账)	准备	114	2%	(114)	100%	114	1 2%	(114)	100%
MARTO 37,5	294,929.32	89.92	4,669,584.20	12.52	32,620,945.12	39,733,347.20	59,49	3,850,592.80	9.69	35,887,154	•											
会计 41.46	69, 203, 66	7	8, 848, 258, 54	1	32, 620, 945, 12	43, 916, 021, 54	1	8, 828, 867, 14	7	35, 887, 154.	•	合计			6,449	100%	(988)	15%	4,853	2 100%	(988)	20%

Fig. 2. Real examples of tables that have similar content but display vastly different visual styles.



Fig. 3. System Structure of the table annotation data synthesis method

within that domain are essential. In the document repository collected from the target domain, layout analysis and table detection are utilized to detect all included tables. OCR or PDF parsing tools are then used to detect and recognize the text content within these tables. Following this, tables are clustered based on their text content, grouping together those with similar content. Subsequently, manual inspection and adjustment are performed. This approach significantly facilitates the convenient and efficient identification of various critical table types within the application domain. Through clustering analysis and adjustment within the Chinese financial announcement document repository, we have identified a comprehensive array of tables. These include financial statements (such as balance sheets, income statements, cash flow statements, statements of shareholders' equity, and supplementary cash flow statements), notes to financial statements (covering accounts receivable, monetary funds, fair value disclosures, and payroll payable), as well as tables detailing directors, supervisors, senior management members, employee statistics, company profiles (encompassing basic information, contact details, principal accounting data, and financial ratios), shareholder information, and glossaries, among others.

We can extract style-related attributes from each table in the repository. These attributes include font information for each text line, line spacing, and alignment within text blocks contained in each cell. Additionally, we consider the alignment of text blocks within cells across rows or columns, the padding between aligned text blocks and their cell bounding boxes, the cells' background color, and the display mode, style, and color of the borders, both external and internal. For detailed illustrations, refer to the Appendix 6.1 and 6.2. These style-related attributes will be stored in the table's style profile, with specific examples provided in the Appendix 6.3.

3.2 Transformation of Table Style

For certain datasets, such as FinTabNet, which already contain table structures and content data in HTML format, we can directly use this HTML data as source material to synthesize new tables through style transformation. In the case of other datasets, like those found in Chinese financial announcement document repositories, although the tables lack directly accessible structure and content information, many are bordered. This characteristic makes it relatively straightforward to deduce the tables' structure by identifying the lines within them and recognizing the text content in each cell. Consequently, these tables can likewise serve as source data for synthesizing tables with new styles.

Based on the content of the current source table, it is possible to match it to a corresponding category. This matching process can be achieved either by training a conventional text classifier or through the application of specific rules, depending on the context. Subsequently, one or more profiles are selected from the Style Profile Candidate Set of the matched category. Minor random adjustments are then made to certain attribute values within the selected profiles, which are ultimately defined as the style profiles for the target tables.

3.3 Synthesis and Rendering of Target Tables

Based on the style profile of the target table, coordinates are calculated for regular non-merged cells using a bottom-up approach (text line \rightarrow text block \rightarrow aligned text block bounding box \rightarrow cell). For illustrations of table elements at various granularities and their corresponding attributes, see Appendices 6.1 and 6.2. Based on the text line attributes defined in the style profile, calculate the height and width of all text lines. If a cell contains multiple text lines, the total height of the text block can be calculated based on the defined line spacing, along with the vertical relative coordinates of the top-left corner of each text line within the text block. Then, by considering the horizontal alignment of the text lines, calculate the total width of the text block and the horizontal relative coordinates of the top-left corner of each text line within the entire text block. Scan each row of the table to identify the tallest text block among all regular non-merged cells and cells merged only in the horizontal direction in the current row. Use this height as the height for the aligned text block bounding box of that row. Calculate the vertical relative coordinates of the text block's top-left corner within the aligned text block bounding box, based on the text block's vertical alignment settings. Then, based on the cell's padding-top and padding-bottom attributes, calculate both the cell's height and the vertical relative coordinates of the aligned text block bounding box within the cell. Similarly, scan each column of the table to identify the widest text block among all regular non-merged cells and cells merged only in the vertical direction in the current column. Use this width as the width for the aligned text block bounding box of that column. Calculate the horizontal relative coordinates of the top-left corner of the text block within the aligned text block bounding box, according to the text block's horizontal alignment settings. Then, using the cell's padding-left and paddingright attributes, determine the cell's width and the horizontal relative positioning of the aligned text block bounding box within the cell.

However, for merged cells or spanning cells, the calculation process is reversed, proceeding from top to bottom (cell \rightarrow aligned text block bounding box \rightarrow text block \rightarrow text line) to compute coordinates. Once the heights of all regular non-merged cells are determined, the height of vertically merged cells can also be obtained. Subsequently, based on the padding-top and padding-bottom attributes, the height of the aligned text block bounding box within this vertically merged cell can be determined. This allows for the calculation of the vertical relative coordinates of the aligned text block bounding box contained in the cell, with the vertical relative coordinates of the text block's top-left corner being calculated based on the text block's vertical alignment settings. Similarly, once the widths of all regular non-merged cells are established, the width of horizontally merged cells can be obtained. Based on the padding-left and padding-right attributes, the width of the aligned text block bounding box within this horizontally merged cell is determined, enabling the calculation of the horizontal relative coordinates of the aligned text block bounding box contained in the cell. The horizontal relative coordinates of the text block's top-left corner within the

aligned text block bounding box are calculated according to the text block's horizontal alignment settings.

Based on the cell sizes calculated in previous steps, the overall size of the table formed by stacking all cells can be determined. By connecting the border lines of cells in the same row and column, the absolute coordinates of the row and column dividers, i.e., the collection of horizontal and vertical lines of the table, are obtained. Calculating the absolute coordinates of the stacked cells, and then based on the relative coordinates of the aligned text block bounding box within the corresponding cell, the relative coordinates of the text block within the aligned text block bounding box, and the relative coordinates of the text block text lines within the corresponding text block, the absolute coordinates of the text lines within the table can ultimately be calculated.

Based on the table size calculated previously, create a blank canvas of the same dimensions. Then, using each cell's coordinates and background color, draw the table's background color. Next, render the corresponding text lines on the image canvas based on the coordinates, color, and font of each text line. Finally, draw the border lines using drawing tools based on the coordinates, mode, line type, and color of the borders. At the same time, output the annotation data in formats required by various table recognition models.

4 Experiments

To verify the practicality and effectiveness of the method proposed in this paper, we conducted experiments with tables from the financial announcement domain. This choice was made because financial announcements contain a large number of complex tables with varying styles. Recognizing these tables not only presents a significant technical challenge but also holds substantial real-world significance for various financial analysis applications.

Financial announcements in the United States have corresponding HTML format documents for their PDF files, and there already exists a large-scale English table annotation dataset like FinTabNet generated through automatic matching. However, Chinese financial announcements in PDF format do not have corresponding structured documents like HTML, and currently, there is no large-scale table annotation dataset available for them. In response to this situation, we have utilized the method proposed in this paper, leveraging the actual structure and content of tables within Chinese financial announcements, to generate the first large-scale table annotation dataset in the domain.

We collected a total of 5049 annual reports from Chinese listed companies in 2022, from which nearly 1.5 million tables were detected and extracted. The majority of these tables are bordered, with a minority being borderless. For comparison purposes, we sampled approximately the same magnitude of tables from these nearly 1.5 million tables as the English financial announcement dataset FinTabNet, totaling 105,600 bordered tables, to serve as the data source for table synthesis. To better recognize more complex tables with a greater number of merged cells, we increased the proportion of challenging complex tables with multiple merged or spanning cells during the sampling process, as shown in Table 1.

Table 1. The data distribution of the 105,600 bordered tables sampled as the data source.

Spanning cell statistics	Number	Percentage
no spanning cell	52208	49.44%
1 spanning cell	4272	4.04%
2 spanning cells	13973	13.23%
3 spanning cells	15582	14.76%
4 spanning cells or more	19565	18.53%

After conducting clustering analysis and manual inspection of the content of these 105,600 bordered tables, they were categorized into 14 categories. Our examination revealed that these source tables already encompass a very broad range of styles found in bordered tables within financial announcements. Therefore, when using these 105,600 bordered tables as the source for synthesizing new tables, 50% of the tables were directly retained as part of the final synthesized table collection as bordered tables, without converting them into different styles of bordered tables. Annotating borderless tables is more challenging than annotating bordered tables. In the synthesized dataset, there is a greater need to enhance support for recognizing borderless tables. Therefore, we transformed the other 50% of the sampled 105,600 bordered tables into borderless tables. The bordered tables serving as source tables can be relatively easily processed by identifying their border lines to extract the table's structure and content. Based on this content, the corresponding category is identified, and then a borderless style profile is selected from the Style Profile Candidate Set of that category. Subsequently, random adjustments of up to 10% are made to certain style attribute values to generate the Style Profile for the synthesized target table. Finally, the target table image is synthesized and rendered, and annotation data is generated.

With the synthesized table annotation dataset available, we can train models for recognizing tables in financial announcements based on it. Here, we reproduced two recent deep learning-based end-to-end models, TableMaster [24] and TableFormer [16], both of which are based on an encoder-decoder architecture, particularly utilizing transformer-based decoders, hence both exhibit strong table recognition performance. Compared to the implementations described in the original papers of TableMaster and TableFormer, we increased the maximum dimension of the input images to 640 pixels to accommodate the recognition of more complex tables. TableMaster and TableFormer have very similar underlying architectures. In TableFormer, the Transformer has fewer layers and heads, and it also employs adaptive pooling to reduce the size of the feature map output by the CNN Backbone. Consequently, TableMaster has a higher number of parameters and, correspondingly, a higher computational complexity. Our experiments also show that TableMaster's overall performance is better

than that of TableFormer. Given that the experimental results and conclusions of both models are consistent, and in the interest of conserving space and presenting the information more concisely, we only report the experimental results of TableMaster in the subsequent sections.

We employ the Tree-Edit-Distance-based Similarity (TEDS) [27], a metric commonly used in table structure recognition literature, to evaluate the performance of table structure recognition. TEDS assesses the similarity between the tree structures of tables. To utilize the TEDS metric, tables must be represented as tree structures in HTML format. Considering that accounting for errors in the text content of tables could result in unfair comparisons due to the varied text extraction methods or OCR models employed by different table recognition methods, we utilize a modified version of TEDS, named TEDS-Struct. This version focuses on the accuracy of table structure recognition, explicitly disregarding the outcomes from text extraction or OCR processes. We also investigate the performance of text block detection (AP50, MS COCO AP at IoU=.50) [9], which is crucial for the precise matching of each cell to its corresponding text content.

We sampled 2,290 real tables from Chinese financial announcements, ensuring no overlap with the 105,600 tables previously sampled as sources for synthesizing table structures and content. Among these, 1,000 are bordered tables, and 1,290 are borderless tables, aiming to increase focus on borderless tables. Additionally, the selection of tables also prioritized more challenging tables that include multiple merged or spanning cells; for specifics, please refer to Table 2. Using table recognition models trained on synthesized data, we conducted recognition and automatic annotation on these 2,290 tables, followed by manual verification to create the first benchmark for complex tables in the Chinese financial announcement domain. This benchmark can be used to evaluate the table recognition performance of models trained on synthesized data, thereby demonstrating the quality of the synthesized data.

Spanning coll statistics	All		Boi	rdered	Borderless		
spanning cen statistics	Number	Percentage	Number	Percentage	Number	Percentage	
No spanning cell	1045	45.63%	442	44.20%	603	46.74%	
1 spanning cell	82	3.58%	59	5.90%	23	1.78%	
2 spanning cells	268	11.70%	145	14.50%	123	9.53%	
3 spanning cells	373	16.29%	100	10.00%	273	21.16%	
4 spanning cells or more	522	22.79%	254	25.40%	268	20.78%	

 Table 2. The data distribution of real-world table benchmark in the Chinese financial announcement domain.

The FinTabNet table dataset, extracted from English financial announcements, was annotated through automatic matching between PDF and HTML, resulting in a considerable number of errors. The FinTabNet training set includes a large number of tables, and due to time constraints, corrections were not conducted. However, for the FinTabNet test set, which contains 10,635 tables, we manually reviewed and used automated scripts to correct 3,733 tables with inconsistent annotations of leader dots. Additionally, we removed 954 tables that had errors in table positioning, structural annotations, or in text box annotations or shifts. The presence of these errors, resulting from automatic annotation, does not imply that the tables are structurally more complex; thus, removing these tables did not decrease the overall difficulty of the FinTabNet table recognition evaluation task. After our review, the corrected FinTabNet test set now comprises 9,681 tables.

Table 3 presents the evaluation results of TableMaster, trained on the synthesized dataset, on our Chinese financial announcement table benchmark dataset. In comparison to the results listed in Table 4—where TableMaster were trained on the FinTabNet training set and evaluated on the corrected FinTabNet test set—the performance is relatively lower. This discrepancy is attributed to the tables extracted from Chinese financial announcements generally being more complex than those from the FinTabNet dataset, which extracts tables from English financial announcements. For instance, tables from Chinese announcements often contain more cells with multi-line text blocks, cells with very long texts, or a higher density of cells. These tables frequently feature structurally complex spanning cells, significant horizontal alignment deviations of text blocks within the same column, among other factors, all of which significantly increase the difficulty of table recognition in Chinese financial announcements. Furthermore, there is still room for improvement in predicting the positions of text block bounding boxes, which could further enhance the accuracy in matching the recognized table structure with the text content [9].

Spanning Cell Statistics	TEDS	TEDS-Struct	AP-50
all tables	0.9091	0.9579	0.482
no spanning cell	0.9216	0.9632	0.501
1 spanning cell	0.9131	0.9503	0.568
2 spanning cells	0.9375	0.9696	0.592
3 spanning cells	0.8785	0.9449	0.505
4 spanning cells or more	0.8908	0.9515	0.458
merged on rows & columns	0.8906	0.9500	0.497

Table 3. The evaluation results for TableMaster, after being trained on the synthesized dataset, on the Chinese financial announcement table benchmark dataset.

To further validate the practicality of our method, we applied the data synthesis method proposed in this paper to augment the training data for FinTab-Net. The original FinTabNet training dataset contains relatively few tables with multiple spanning cells. In augmenting the dataset while keeping the total number of tables roughly the same, a portion of the augmented dataset was obtained directly by sampling from the original FinTabNet training dataset, and another portion was synthesized using the method described in this paper. The aug-

Table 4.	The evaluation	on results for	• TableMaster,	after being	trained on	the FinTabNe
training s	set, on the co	rrected FinT	abNet test set	.		

Spanning cell statistics	TEDS	TEDS-Struct	AP-50
all tables	0.9758	0.9856	0.619
no spanning cell	0.9727	0.9829	0.631
1 spanning cell	0.9830	0.9922	0.654
2 spanning cells	0.9818	0.9896	0.630
3 spanning cells	0.9657	0.9783	0.557
4 spanning cells or more	0.9342	0.9552	0.511
merged on rows & columns	0.9503	0.9620	0.574

mented dataset increases the proportion of tables containing multiple spanning cells, with specific distribution information of the tables available in Table 5. Table 6 presents the experimental results on the corrected FinTabNet test dataset of models trained using the augmented FinTabNet training data. Compared with Table 4, it shows that data augmentation using the method proposed in this paper can comprehensively improve the performance of FinTabNet table recognition, especially in recognizing more complex tables with multiple spanning cells. This demonstrates the effectiveness of our table synthesis method for practical application scenarios.

 Table 5. The data distribution of Augmented FinTabNet training dataset.

Table Type	FinTabNot	Augmented dataset				
Table Type	I III I abivet	Sampled	Synthesized	Sum		
no spanning cell	44k+	10000	10000	20000		
1 spanning cell	22k+	7500	7500	15000		
2 spanning cells	14k+	14439	561	15000		
3 spanning cells	4k+	4824	10176	15000		
4 spanning cells or more	5k+	5026	14711	19737		
merged on rows & columns	2k+	263	7954	8217		

5 Conclusion and Future Work

Unlike previous methods that rely on automatic matching between PDFs and structured source codes or on random synthesis of table structures and content, this paper introduces a novel approach for synthesizing high-quality tables and annotated data. This method leverages the structure and content of existing real tables to replicate authentic table styles of the target domain. We applied this approach within the financial sector, producing the first extensive table annotation dataset for the Chinese financial announcement domain and enhancing the English financial table dataset, FinTabNet. Our experiments demonstrate the real-world applicability and effectiveness of this table synthesis method.

Spanning cell statistics	TEDS	TEDS-Struct	AP-50
all tables	0.9847	0.9971	0.789
no spanning cell	0.9805	0.9971	0.746
1 spanning cell	0.9895	0.9982	0.813
2 spanning cells	0.9912	0.9979	0.827
3 spanning cells	0.9794	0.9906	0.809
4 spanning cells or more	0.9740	0.9906	0.747
merged on rows & columns	0.9684	0.9794	0.841

Table 6. The evaluation results for TableMaster, after being trained on the Augmented

 FinTabNet training set, on the corrected FinTabNet test set.

The experiments utilized end-to-end table recognition methods such as Table-Master, chosen for their relatively simple data preparation process and proven excellence in performance across previous studies. However, these methods encounter considerable challenges when recognizing the structurally complex tables common in Chinese financial announcements. Future work will explore additional methods, particularly those based on segmentation and merging [21], which are anticipated to yield improved results for tables with intricate spanning cell structures. Employing a diverse range of table recognition methods will enable a more thorough assessment of the synthesized data's quality.

Furthermore, we aim to broaden the diversity of table styles by incorporating tables from audit reports, to publicly release our extensive synthesized dataset for the Chinese financial announcement domain, and to enlarge and publicly unveil the benchmark for real complex tables within the same domain.

References

- Chen, L., Huang, C., Zheng, X., Lin, J., Huang, X.J.: Tablevlm: Multi-modal pretraining for table structure recognition. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 2437–2449 (2023)
- Deng, Y., Rosenberg, D., Mann, G.: Challenges in end-to-end neural scientific table recognition. In: 2019 International Conference on Document Analysis and Recognition (ICDAR). pp. 894–901 (2019)
- Desai, H., Kayal, P., Singh, M.: Tablex: a benchmark dataset for structure and content information extraction from scientific tables. In: Document Analysis and Recognition–ICDAR 2021: 16th International Conference, Lausanne, Switzerland, September 5–10, 2021, Proceedings, Part II 16. pp. 554–569. Springer (2021)
- Duda, R.O., Hart, P.E.: Use of the hough transformation to detect lines and curves in pictures. Commun. ACM 15(1), 11–15 (jan 1972)
- Grompone von Gioi, R., Jakubowicz, J., Morel, J.M., Randall, G.: LSD: a Line Segment Detector. Image Processing On Line 2, 35–55 (2012)
- Gu, G., Ko, B., Go, S., Lee, S.H., Lee, J., Shin, M.: Towards light-weight and realtime line segment detection. Proceedings of the AAAI Conference on Artificial Intelligence 36(1), 726–734 (Jun 2022)

- 16 Q. Hou et al.
- Huang, K., Wang, Y., Zhou, Z., Ding, T., Gao, S., Ma, Y.: Learning to parse wireframes in images of man-made environments. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018)
- Huang, S., Qin, F., Xiong, P., Ding, N., He, Y., Liu, X.: Tp-lsd: Tri-points based line segment detector. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.M. (eds.) Computer Vision – ECCV 2020. pp. 770–785. Springer International Publishing, Cham (2020)
- Huang, Y., Lu, N., Chen, D., Li, Y., Xie, Z., Zhu, S., Gao, L., Peng, W.: Improving table structure recognition with visual-alignment sequential coordinate modeling. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11134–11143. IEEE Computer Society, Los Alamitos, CA, USA (jun 2023)
- Ichikawa, K.: Image-based relation classification approach for table structure recognition. In: Document Analysis and Recognition – ICDAR 2021: 16th International Conference, Lausanne, Switzerland, September 5–10, 2021, Proceedings, Part II. p. 632–647. Springer-Verlag, Berlin, Heidelberg (2021)
- Khan, U., Zahid, S., Ali, M.A., Ul-Hasan, A., Shafait, F.: Tabaug: data driven augmentation for enhanced table structure recognition. In: Document Analysis and Recognition–ICDAR 2021: 16th International Conference, Lausanne, Switzerland, September 5–10, 2021, Proceedings, Part II 16. pp. 585–601. Springer (2021)
- Li, M., Cui, L., Huang, S., Wei, F., Zhou, M., Li, Z.: TableBank: Table benchmark for image-based table detection and recognition. In: Proceedings of the Twelfth Language Resources and Evaluation Conference. pp. 1918–1925. European Language Resources Association, Marseille, France (May 2020)
- Liu, H., Li, X., Liu, B., Jiang, D., Liu, Y., Ren, B.: Neural collaborative graph machines for table structure recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4533–4542 (2022)
- Ly, N., Takasu, A., Nguyen, P., Takeda, H.: Rethinking image-based table recognition using weakly supervised methods. In: Proceedings of the 12th International Conference on Pattern Recognition Applications and Methods. SCITEPRESS -Science and Technology Publications (2023)
- Matas, J., Galambos, C., Kittler, J.: Robust detection of lines using the progressive probabilistic hough transform. Computer Vision and Image Understanding 78(1), 119–137 (2000)
- Nassar, A., Livathinos, N., Lysak, M., Staar, P.: Tableformer: Table structure understanding with transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4614–4623 (June 2022)
- Qasim, S.R., Mahmood, H., Shafait, F.: Rethinking table recognition using graph neural networks. In: 2019 International Conference on Document Analysis and Recognition (ICDAR). pp. 142–147. IEEE (2019)
- Qiao, L., Li, Z., Cheng, Z., Zhang, P., Pu, S., Niu, Y., Ren, W., Tan, W., Wu, F.: Lgpma: Complicated table structure recognition with local and global pyramid mask alignment. In: Lladós, J., Lopresti, D., Uchida, S. (eds.) Document Analysis and Recognition – ICDAR 2021. pp. 99–114. Springer International Publishing, Cham (2021)
- Raja, S., Mondal, A., Jawahar, C.V.: Table structure recognition using top-down and bottom-up cues. In: Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVIII. p. 70–86. Springer-Verlag, Berlin, Heidelberg (2020)

17

- Smock, B., Pesala, R., Abraham, R.: Pubtables-1m: Towards comprehensive table extraction from unstructured documents. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4634–4642 (June 2022)
- Tensmeyer, C., Morariu, V.I., Price, B., Cohen, S., Martinez, T.: Deep splitting and merging for table structure decomposition. In: 2019 International Conference on Document Analysis and Recognition (ICDAR). pp. 114–121 (2019)
- Umer, M., Mohsin, M.A., Ul-Hasan, A., Shafait, F.: Pyramidtabnet: Transformerbased table recognition in image-based documents. In: Fink, G.A., Jain, R., Kise, K., Zanibbi, R. (eds.) Document Analysis and Recognition - ICDAR 2023. pp. 420–437. Springer Nature Switzerland, Cham (2023)
- Xu, Y., Xu, W., Cheung, D., Tu, Z.: Line segment detection using transformers without edges. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4255–4264. IEEE Computer Society, Los Alamitos, CA, USA (jun 2021)
- Ye, J., Qi, X., He, Y., Chen, Y., Gu, D., Gao, P., Xiao, R.: Pingan-vcgroup's solution for icdar 2021 competition on scientific literature parsing task b: Table recognition to html (2021)
- Zhang, Z., Li, Z., Bi, N., Zheng, J., Wang, J., Huang, K., Luo, W., Xu, Y., Gao, S.: Ppgnet: Learning point-pair graph for line segment detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2019)
- Zheng, X., Burdick, D., Popa, L., Zhong, X., Wang, N.X.R.: Global table extractor (gte): A framework for joint table identification and cell structure recognition using visual context. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 697–706 (January 2021)
- Zhong, X., ShafieiBavani, E., Jimeno Yepes, A.: Image-based table recognition: Data, model, and evaluation. In: Computer Vision – ECCV 2020. pp. 564–580. Springer International Publishing, Cham (2020)
- Zhou, W.: Tablegeneration. https://github.com/WenmuZhou/TableGeneration (2022)
- Zhou, Y., Qi, H., Ma, Y.: End-to-end wireframe parsing. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 962–971 (2019)

6 Appendix

6.1 Table Elements at various granularities

Cells In Fig 4, each area filled with a different color represents a cell. Only tables with complete border lines have definite cell coordinates.

Text Lines In Fig. 5, the areas filled with color represent text lines. Text lines within the same cell share the same color, and a single cell often contains multiple text lines.

Text Blocks In Fig. 6, each area filled with a different color represents a text block, and the bounding box of a text block covers all text lines within the same cell.

			上年年末余额		
26 11	账面余额		坏账准		
类别		比例	6 200	计提比	账面价值
	金額	(%)	金额	例(%)	
单项金额重大并单独					
计提坏账准备的应收					
账款					
按信用风险特征组合					
计提坏账准备的应收	565,198,202.06	98.32	33,911,892.13	6.00	531,286,309.93
账款					
单项金额不重大但单					
独计提坏账准备的应	9,632,677.56	1.68	9,632,677.56	100.00	
收账款					
合计	574,830,879.62	100.00	43,544,569.69		531,286,309.93

Fig. 4. Illustration of table cells.

			上年年末余额			
NETT	账面余额		坏账准			
类别		比例		计提比	账面价值	
	金额	(%)	金额	例(%)		
单项金额重大并单独						
计提坏账准备的应收						
账款						
按信用风险特征组合						
计提坏账准备的应收	565,198,202.06	98.32	33,911,892.13	6.00	531,286,309.93	
账款						
单项金额不重大但单						
独计提坏账准备的应	9,632,677.56	1.68	9,632,677.56	100.00		
收账款						
合计	574,830,879.62	100.00	43,544,569.69		531,286,309.93	

Fig. 5. Illustration of text lines.



Fig. 6. Illustration of text blocks.

Aligned Text Block Bounding Boxes In Fig. 7, each area filled with a different color is an aligned text block bounding box. Identify the highest upper bound and the lowest lower bound from all text blocks among all regular nonmerged cells and cells merged only in the horizontal direction in the current row to calculate the height, and use this height as the height of the aligned text block bounding box for that row. Identify the leftmost boundary and the rightmost boundary from all text blocks among all regular non-merged cells and cells merged only in the vertical direction in the current column to calculate the width, and use this width as the width of the aligned text block bounding box for that column. The width of the aligned text block bounding box for a horizontally merged cell is determined by the left boundary of the aligned text block bounding box of the leftmost column involved in the merge and the right boundary of the aligned text block bounding box of the rightmost column involved in the merge. The height of the aligned text block bounding box for a vertically merged cell is determined by the upper boundary of the aligned text block bounding box of the topmost row involved in the merge and the lower boundary of the aligned text block bounding box of the bottommost row involved in the merge.

6.2 Attributes of Table Elements

Attributes of Text Lines and Text Blocks The attributes of text lines include the font, size, and color of the text, all of which can be set at various granularities such as table, row, column, or cell. The attributes of text blocks include the line spacing between the contained text lines and the alignment of text lines within the text block (centered/left-aligned/right-aligned/specified



Fig. 7. Illustration of Aligned Text Block Bounding Boxes.

indent distance), all of which can be set at various granularities such as table, row, column, or cell. Fig. 8 shows an illustration of the attributes of a text block and the text lines contained within it, extracted from Fig. 6.



Fig. 8. Illustration of Attributes of text lines and text blocks.

Attributes of Cells and corresponding Aligned Text Block Bounding Boxes The attributes of an aligned text block bounding box mainly involve the alignment of the contained text block, including horizontal alignment (leftaligned, right-aligned, centered, indented by a specified distance to the left or right) and vertical alignment (top-aligned, bottom-aligned, centered, indented by a specified distance from the top or bottom). All of these can be set at various granularities such as table, row, column, or cell. The attributes of a cell include the padding, defined as the distance between the cell bounding box and the aligned text block bounding box, as well as the cell's background color. All of these can be set at various granularities such as table, row, column, or cell. Fig. 9 displays a cell and its corresponding aligned text block bounding box, as extracted from Fig. 7.



Fig. 9. Illustration of Attributes of Cells and corresponding Aligned Text Block Bounding Boxes.

Attributes of borders In tables, borders are categorized into outer and inner borders. The attributes of outer borders apply to the table as a whole, while inner border attributes can be adjusted at various levels of granularity, such as by table, row, column, or cell. Outer border modes can be fully visible, absent on the sides, absent on the top and bottom, or completely absent. Inner border options encompass fully visible, absent horizontal lines, absent vertical lines, completely absent, partially absent horizontal lines, and partially absent vertical lines. Outer border line types include options like single solid lines and double solid lines, along with specifications for their thickness. Inner border line types feature solid lines, dashed lines, and details regarding line thickness and the spacing in dashed lines. Furthermore, colors can be specified for both outer and inner borders.

6.3 Table Style Profile

Fig. 10 shows an example of the format of a Table Style Profile.



Fig. 10. Illustration of an example of the format of a Table Style Profile.