




GFTE: Graph-Based Financial Table Extraction

Yiren Li¹, Zheng Huang¹, Junchi Yan¹, Yi Zhou¹, Fan Ye²,
and Xianhui Liu²

¹ School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai, China

{irene716, huang-zheng, yanjunch, izy_21th}@sjtu.edu.cn

² China Financial Fraud Research Center, Shanghai, China
yefan@xnai.edu.cn, lxianhui@tjzxcn.com

Abstract. Tabular data is a crucial form of information expression, which can organize data in a standard structure for easy information retrieval and comparison. However, in financial industry and many other fields, tables are often disclosed in unstructured digital files, e.g. Portable Document Format (PDF) and images, which are difficult to be extracted directly. In this paper, to facilitate deep learning based table extraction from unstructured digital files, we publish a standard Chinese dataset named FinTab, which contains more than 1,600 financial tables of diverse kinds and their corresponding structure representation in JSON. In addition, we propose a novel graph-based convolutional neural network model named GFTE as a baseline for future comparison. GFTE integrates image feature, position feature and textual feature together for precise edge prediction and reaches overall good results <https://github.com/Irene323/GFTE>.

Keywords: Deep learning · Document analysis · Document image processing

1 Introduction

In the information age, how to quickly obtain information and extract key information from massive and complex resources has become an important issue [1]. In the meantime, with the increase in the number of enterprises and the growing amount of disclosure of financial information, extracting key information has also become an essential means to improve the efficiency in financial information exchange process. In recent years, some new researches have also begun to focus on improving the efficiency and accuracy of information retrieval technology [2, 3].

Table, as a form of structured data, is both simple and standardized. Hurst et al. [4] regard a table as a representation of a set of relations between organized hierarchical concepts or categories, while Long et al. [5] consider it as a super-structure imposed on a character-level grid. Due to its clear structure, table data

can be quickly understood by users. Financial data, especially digital information, are often presented in tabular form. In a manner of speaking, table data, as key information in financial data, are increasingly valued by financial workers during financial data processing.

Table data in finance context have entirely different characteristics from ordinary table data in daily life or in academia:

1. The application of table data is widespread.
2. Complex table structure is difficult to be extracted.

Tables in financial data have various sources and forms. Thus, the structure of financial table can be rather complicated. For example:

- Some financial documents use various tables without complete table lines for elegant typography.
- Many cells in financial table are merged to indicate values in different categories or in different stages.
- There are often a lot of information in the cells of financial tables. For example, a cell may contain a large number that consists of multiple digits, or it may have many digits after the decimal point.
- There are also cases when one cell holds a lot of text information. This may lead to the division of one single cell into two pages, especially when it is located at the end of one page.
- One financial table may store hundreds of thousands of cells, which will occupy multiple consecutive pages.

The results extracted from these complex tables often have data confusion or overlap.

3. Financial data demand high quality and accuracy.

Although table extraction is a common task in various domains, extracting tabular information manually is often a tedious and time-consuming process. We thus require automatic table extraction methods to avoid manual involvement. However, it is still difficult for the existing methods to accurately recover the structure of relatively complicated financial tables.

Figure 1 illustrates an intuitive example of the performance of different existing methods, i.e. Adobe Acrobat DC and Tabby [6]. Both of them fail to give the correct result. Meanwhile, it is not hard to notice that problems often occur at spanning cells, which very likely carry the information of table headers and are thus critical for table extraction and understanding. Therefore, the performance of table extraction methods are still hoped to be improved, especially by the complicated cases.

Based on these considerations, since the design of artificial intelligence algorithms relies on standard data and test benchmarks, we construct an open source financial benchmark dataset named FinTab. More specifically, sample collection, sample sorting and cleaning, benchmark data determining and baseline method test were completed. FinTab can be further used in financial context in terms of table extraction, key information extraction, image data identification, bill identification and other specific content. With a more comprehensive benchmark dataset, we hope to promote the emergence of more innovative technologies.

Year 2002	Return on equity		Earnings per share (yuan)	
	Fully diluted	Weighted average	Fully diluted	Weighted average
Income from main operation*	264%	287%	3.15	3.15

(a) The ground truth structure of part of a financial table. Each cell in the first two lines is marked with different colors.

Return on equity	Earnings				
per share (yuan)					
Year 2002	Fully diluted	Weighted average	Fully diluted	Weighted average	
Income from main operation*	264%	287%	3.15	3.15	

(b) The recognized structure by Adobe Acrobat DC.

	Return on equity		Earnings per share (yuan)	
Year 2002				
	Fully diluted	Weighted average	Fully diluted	Weighted average
Income from main operation*	264%	287%	3.15	3.15

(c) The recognized structure using Tabby [6].

Fig. 1. An example of a table with spanned cells and the recovered table structures with the existing methods.

Further detailed information about our standard financial dataset will be introduced in Sect. 3.

Besides, this paper also proposes a novel table extraction method, named GFTE, with the help of graph convolutional network (GCN). GFTE can be used as a baseline, which regards the task of table structure recognition as an edge prediction problem based on graph. More specifically, we integrate image feature, textual feature and position feature together and feed them to a GCN to predict the relation between two nodes. Details about this baseline algorithm will be discussed in Sect. 4.

In general, the contributions of this work can be summarized as following:

1. A Chinese benchmark dataset FinTab of more than 1,600 tables of various difficulties, containing table location, structure identification and table interpretation information.
2. We propose a graph-based convolutional network model named GFTE as table extraction baseline. Extensive experiments demonstrate that our proposed model outperforms state-of-the-art baselines greatly.

2 Related Work

In this section, we will first familiarize the reader with some previous published datasets and some related contests, and then present a overview of table extraction technologies.

2.1 Previous Datasets

We introduce some existing public available datasets:

- The Marmot dataset [7] is composed of both Chinese and English pages. The Chinese pages are collected from over 120 e-Books in diverse fields of subject provided by Founder Apabi library, while the English pages are from Citeseer website. Derived from PDF, the dataset stores tree structure of all document layouts, where the leaves are characters, images and paths, and the root is the whole page. Internal nodes include textlines, paragraphs, tables etc.
- The UW3 dataset [8] is collected from 1,600 pages of skew-corrected English document and 120 of them contain at least one marked table zone. The UNLV dataset derives from 2,889 pages of scanned document images, in which 427 images include table.
- The ICDAR 2013 dataset [9] includes a total of 150 tables: 75 tables in 27 excerpts from the EU and 75 tables in 40 excerpts from the US Government, i.e. in total 67 PDF documents with 238 pages in English.
- This dataset for the ICDAR 2019 Competition on Table Detection and Recognition [10] is separated into training part and test part. The training dataset contains images of 600 modern documents and their bounding boxes of table region, as well as images of 600 archival documents, their table structures and bounding boxes of both table region and cell region. In the test dataset, images and table regions of 199 archival documents and 240 modern ones are offered. Besides, table structures and cell regions of 350 archival documents are also included.
- The PubTabNet dataset [11] contains more than 568 thousand images of tabular data annotated with the corresponding HTML representation of the tables. More specifically, table structure and characters are offered but the bounding boxes are missing.
- The SciTSR dataset [12] is a comprehensive dataset, which consists of 15,000 tables in PDF format, images of the table region, their corresponding structure labels and bounding boxes of each cell. It is split into 1,200 for training and 3,000 for test. Meanwhile, a list of complicated tables, called SciTSR-COMP, is also provided.
- The TableBank [13] is an image-based table detection and recognition dataset. Since two tasks are involved, it is composed of two parts. For the table detection task, images of the pages and bounding boxes of tables region are included. For the table structure recognition task, images of the page and HTML tag sequence that represents the arrangement of rows and columns as well as the type of table cells are provided. However, textual content recognition is not the focus of this work, so textual content and its bounding boxes are not contained.

Table 1 gives more information for comparison.

2.2 Methods

Table extraction is considered as a part of table understanding [14], and conventionally consists of two steps [6]:

Table 1. Public datasets for table recognition

Name	Source	Content	Amount	Language
Marmot	e-Books and Citeseer website	Tree structure of all document layouts; bmp, xml	2,000	Chinese, English
UW3	Skew-corrected document	Images of page, manually edited bounding boxes of page frame, text and non-text zones, textlines, and words, type of each zone; png, xml	120	English
UNLV	Magazines, news papers, business letter, annual Report, etc.	Scanned images of page, bounding boxes for rows, columns and cells, OCR recognized words within the whole page; png, xml	427	English
ICDAR 2013	European Union and US Government websites	PDF documents, bounding boxes of table, textual content, structure labels, bounding boxes for each cell; pdf, xml	150	English
ICDAR 2019	Modern documents and archival ones with various formats	840 jpgs and xmls including bounding boxes of table in modern documents, 1149 jpgs and xmls including structure labels, bounding boxes of table and each cell in archival documents	About 2,000	English
PubTabNet	Scientific publications included in the PubMed Central Open Access Subset	Images of tabular data, textual content, table structure labels in HTML; png, json	More than 568,000	English
SciTSR	LaTeX source files	PDF, images, textual content, structure labels, bounding boxes for each cell	15,000	English
TableBank	Word and Latex documents on the internet such as official fillings, research papers, etc.	Table Detection: bounding boxes of table; Table Structure Recognition: table structure labels; jpg, HTML	417,234; 145,463	English
FinTab	Annual and semi-annual reports, debt financing, bond financing, collection of medium-term notes, short-term financing, prospectus	Textual ground truth, structure information and the unit of the table; pdf, json	1,685	Chinese

1. Table Detection. Namely, a certain part of the file is identified as a table in this step.
2. Table Structure Decomposition. This task aims to recover the table into its components as close to the original as possible. For example, the proper identification of header elements, the structure of columns and rows, correct allocation of data units, etc.

In the past two decades, a few methods and tools have been devised for table extraction. Some of them are discussed and compared in some recent surveys [15, 16].

There are generally three main categories in the existing approaches [17]: Predefined layout-based approaches, Heuristic-based approaches and Statistical or optimization-based approaches.

Predefined layout-based approaches design several templates for possible table structures. Certain parts of the document is identified as tables, if they correspond to certain templates. Shamilian [18] proposes a predefined layout-based table identification and segmentation algorithm as well as a graphical user interface (GUI) for defining new layouts. Nevertheless, it only works well in single-column cases. Mohemad et al. [19] present another predefined layout-based approach, which focuses on paragraph and tabular, then associated text using a combination of heuristic, rule-based and predefined indicators. However, a disadvantage of these approaches is that tables can only be classified into the previous defined layouts, while there are always limited types of templates defined in advance.

Heuristic-based approaches specify a set of rules to make decisions so as to detect tables which meet certain criteria. According to [16], heuristics-based approaches remain dominant in literature. [20] is the first relative research focusing on PDF table extraction, which uses a tool named pdf2html to return text pieces and their absolute coordinates, and then utilizes them for table detection and decomposition. This technique achieves good results for lucid tables, but it is limited as it assumes all pages to be single column. Liu et al. [21] propose a set of medium-independent table metadata to facilitate the table indexing, searching, and exchanging, in order to extract the contents of tables and their metadata.

Statistical approaches make use of statistical measures obtained through offline training. The estimated parameters are then taken for practical table extraction. Different statistical models have been used, for example, probabilistic modelling [22], the Naive Bayes classifier [23, 24], decision trees [25, 26], Support Vector Machine [25, 27], Conditional Random Fields [27–29], graph neural network [12, 30, 31], attention module [32], etc. [33] uses a pair of deep learning models (Split and Merge models) to recover tables from images.

3 Dataset Collection

In general, there are currently following problems with the existing contests and standard datasets:

1. There are few competitions and standard datasets for extracting table information from financial documents.
2. The source for tabular information extraction lacks diversity.

In consideration of this, the benchmark dataset FinTab released this time aims to make certain contribution in this field. In this dataset, we collect a total of 19 PDF files with more than 1,600 tables. The specific document classification is shown in Table 2. All documents add up to 3,329 pages, while 2,522 of them contain tables.

Table 2. Document classification of our benchmark dataset FinTab

File type	Number of files
Annual and semi-annual reports	3
Debt financing	2
Bond financing	3
Collection of medium-term notes	2
Short-term financing	8
Prospectus	1

FinTab provides more comprehensive details of the table than any other datasets introduced in Sect. 2. It is also worth noticing that FinTab has been manually reviewed, which makes it much more reliable. We provide both characters and strings as textual ground truth. For structure ground truth of a table, we present the detailed information of its cells and its table lines. More specifically, different kinds of ground truth of a table are stored in json files as shown in Table 3.

Table 3. Ground truth provided in FinTab

Information type	Ground truth included
Table position	4 coordinates of the bounding box
Table line	2 coordinates of the start point and the end point, line color, fill color
Character	Textual content, font size, font name, font color, 4 coordinates of the bounding box
String	Textual content, font size, font color, 4 coordinates of the bounding box
Cell	Structure position in the table, 4 coordinates of the bounding box

To ensure that the types of forms are diverse, in addition to the basic forms of table, special cases with different difficulties are also included, e.g. semi-ruled table, cross-page table, table with merged cells, multi-line header table, etc. It is also worth mentioning that there are 119,021 cells in total, while the number of merged cells is 2,859, accounting for 2.4%. Detailed types and quantity distribution of tables are shown in Table 4.

Table 4. Document classification of our benchmark dataset FinTab

Table type	Number of tables	Percentage
Single-page table	523	62.5%
Double-page table	523	31%
Multi-page table	108	6.5%
Table with incomplete form line	140–150	8.3%-8.9%
Table with merged cells	583	34.59%
Total	1,685	100%

FinTab contains various types of tables. Here, we briefly introduce some of them in order of difficulty.

1. Basic single-page table. This is the most basic type of table, which takes up less than one page and does not include merged cells. It is worth mentioning that we offer not only textual ground truth and structure information, but also the unit of the table, because mostly financial table contains quite a few numbers.
2. Table with merged cells. In this case, the corresponding merged cells should be recovered.
3. Cross-page table. If the table appears to spread across pages, the cross-page table need to be merged into a single form. If the header of the two pages appears to be duplicated, only one needs to be remained. Page number and other useless information should also be removed. Another difficult situation to be noticed is that if a single cell is separated by two pages, it should be merged into one according to its semantics.
4. Table with incomplete form line. In this case, it is necessary to intelligently locate the dividing line according to the position, format, and meaning of the text.

4 Baseline Algorithm

In this paper, we also propose a novel graph-neural-network-based algorithm named GFTE to fulfill table structure recognition task, which can be used as a baseline. In this section, we introduce detailed procedure of this algorithm.

Figure 2 illustrates an overview of GFTE. Since our dataset is in Chinese, we give a translated version of the example in Table 5 for better understanding. To first train our model, the following steps are carried out on the training dataset:

- a. Given a certain table, we load its ground truth, which consists of (1) image of the table region, (2) textual content, (3) text position and (4) structure labels.
- b. Then, based on the ground truth (1)-(3) we construct an undirected graph $G = \langle V, R_C \rangle$ on the cells.
- c. After that, we use our GCN-based algorithm to predict adjacent relations, including both vertical relations (namely whether two nodes are in the same row) and horizontal relations (namely whether two nodes are in the same column).
- d. By comparing the prediction with ground truth (4), i.e. the structure labels, we can calculate the loss and optimize the model.

After the model is trained to a satisfactory level, given an image of a certain table, we should be able to recognize the strings and their position in the image. Then, our GFTE model would predict the relationship between these strings and finally recover the structure of the table.

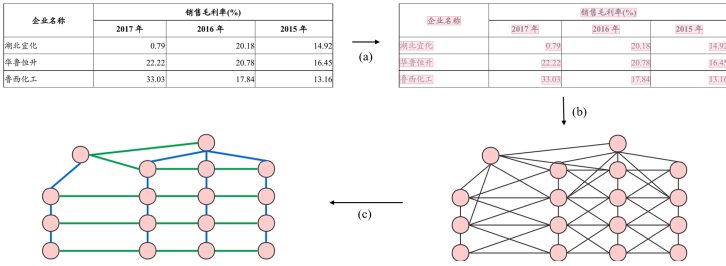


Fig. 2. Overview of our novel GCN-based algorithm.

Table 5. The translated version of the table we used for illustration in this paper.

Company name	Gross profit margin(%)		
	Year 2017	Year 2016	Year 2015
Hubei Yihua	0.79	20.18	14.92
Hualu Hengsheng	22.22	20.78	16.45
Luxi Chemical	33.03	17.84	13.16

In the next sub-section, we first introduce how we comprehend this table structure recognition problem.

Problem Interpretation. In a table recognition problem, it is quite natural to consider each character string in the table as a node. Then, the vertical or horizontal relation between a node and its neighbors can be understood as the feature of edges. More specifically, for a particular node, the vertical relation can be considered as “exist” only on the edges between this node and other nodes in the same column. Similarly, for this particular node, the horizontal relation only exists on edges between this node and other nodes in the same row.

If we use N to denote the set of nodes and E_C to denote the fully connected edges, then a table structure can be represented by a complete graph $G = \langle V, R_C \rangle$, where R_C indicate a set of relations between E_C . More specifically, we have $R_C = E_C \times \{vertical, horizontal\}$.

Thus, we can interpret the problem as the following: given a set of nodes N and their feature, our aim is to predict the relations R_C between pairs of nodes as accurate as possible.

However, training on complete graphs is expensive. It is not only computationally intensive but also quite time-consuming. Meanwhile, it is not hard to notice that a table structure can be represented by far fewer edges, as long as a node is connected to its nearest neighbors including both vertical ones and horizontal ones. With the knowledge of node position, we are also capable of recovering the table structure from these relations.

Therefore in this paper, instead of training on the complete graph with R_C , which is of $O(|N|^2)$ complexity, we make use of the K-Nearest-Neighbors (KNN) method to construct R , which contains the relations between each node and its K nearest neighbors. With the help of KNN, we can reduce the complexity to $O(K * |N|)$.

GFTE. For each node, three types of information are included, i.e. the textual content, the absolute locations and the image, as shown in Fig. 3. We then make use of the structure relations to build the ground truth and the entire structure could be like Fig. 4. For higher accuracy, we train horizontal and vertical relations separately. For horizontal relations, we label each edge as 1: *in the same row* or 0: *not in the same row*. Similarly for vertical relations, we label each edge as 1: *in the same column* or 0: *not in the same column*.

Figure 5 gives the structure of our graph-based convolutional network GFTE. We first convert the absolute position into relative positions, which are further used to generate the graph. In the mean time, plain text is first embedded into a predefined feature space, then LSTM is used to obtain semantic feature. We concatenate the position feature and the text feature together and feed them to a two-layer graph convolutional network (GCN).

Meanwhile, we first dilate the image by a small kernel to make the table lines thicker. We also resize the image to 256×256 pixels in order to normalize the input. We then use a three-layer CNN to calculate the image feature. After that, using the relative position of the node, we can calculate a flow-field grid. By computing the output using input pixel locations from the grid, we can acquire the image feature of a certain node at a certain point.

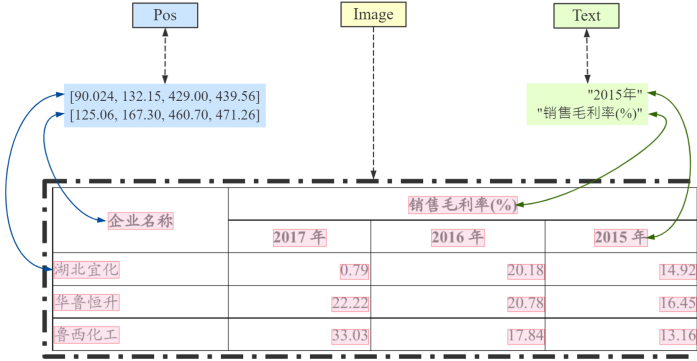


Fig. 3. An intuitive example of source data format.

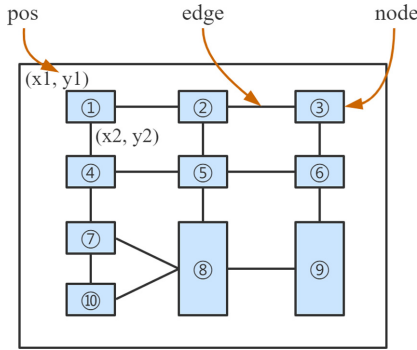


Fig. 4. Ground truth structure.

When these three different kinds of features are prepared, we pair two nodes on an edge of the generated graph. Namely we find two nodes of one edge and concatenate their three different kinds of features together. Finally, we use MLP to predict whether the two nodes are in same row or in same column.

5 Evaluation Results

In this section, we evaluate GFTE with prediction accuracy, as used in [31], for both vertical and horizontal relations. Our novel FinTab dataset is separated into train part and test part and is used to evaluate the performance of different GFTE model structures, meanwhile the SciTSR dataset is also used for validation.

Firstly, we train GFTE-pos. Namely we use the relative position and KNN algorithm to generate graph, and we train GFTE only with the position feature. Secondly, we train the network with the position feature as well as the text feature acquired by LSTM. This model is named GFTE-pos+text. Finally, our

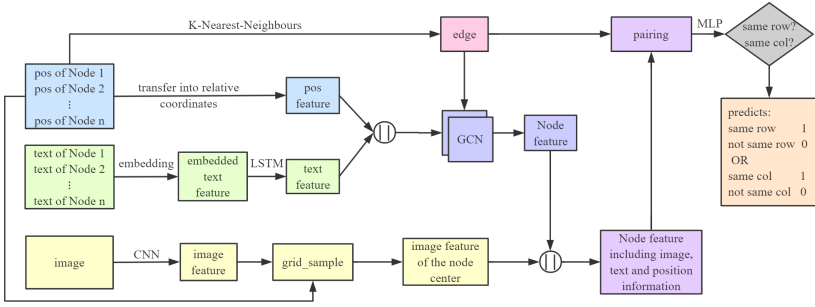


Fig. 5. The structure of our proposed GCN-based algorithm GFTE.

proposed GFTE is trained by further including the image feature with the help of grid sampling.

In Table 6, we give the performance of different models on FinTab dataset. As listed, the accuracy shows an overall upward trend when we concatenate more kinds of features. It improves distinctly when we include text feature, namely a rise of 10% by horizontal prediction and 5% by vertical prediction. Further including image feature seems to help improve the performance a little, but not too much.

Table 6. Accuracy results of different GFTE models on vertical and horizontal directions.

Network	Horizontal prediction	Vertical prediction
GFTE-pos	0.759836	0.842450
GFTE-pos+text	0.858675	0.903230
GFTE	0.861019	0.903031

Meanwhile, we notice a higher accuracy in vertical prediction than in horizontal prediction on FinTab. It is possibly caused by the uneven distribution of cells within a row of financial tables. Figure 6 gives some typical examples. In Fig. 6 (a), the nodes in the first 8 rows are distributed extremely far in horizontal direction. In Fig. 6 (b), when calculating K nearest neighbors for the first column, many vertical relations will be included, but very few horizontal relations, especially when K is small. These situations are rather rare in academic tables but not uncommon in financial reports.

In Table 7, we give the accuracy results of GFTE on different datasets, namely on the SciTSR test dataset and on our FinTab test dataset. It could be observed that our model reaches rather high accuracy on SciTSR validation dataset, which

名称 (全名)		Development Principles Fund II L.P.	
成立时间		2008 年 3 月 14 日	
合伙人认缴出资		101,025,001.00 美元	
合伙人实际缴纳资本		92,589,110.00 美元	
注册地		开曼群岛	
主要生产经营地		开曼群岛	
合伙类型		有限合伙	
合伙期限		10 年	
合伙人姓名 或名称及认 缴资本额	有限合伙人	Nederlandse Financierings-Maatschappij voor Ontwikkelingslanden N.V.	4,000 万美元
		DEG-Deutsche Investitions-und Entwicklungsgesellschaft MBH	2,000 万美元
		Société de Promotion et de Participation pour la Coopération Économique	2,000 万美元
		ITOCHU Finance (Asia) Limited	1,000 万美元
		ITOCHU Hong Kong Limited	1,000 万美元
		The Development Principles Group Limited	102.5 万美元
		Development Principles Fund II CIP Limited	1.0 美元
	普通合伙人	Development Principles Fund II GP Ltd.	10,000 美元
执行事务合伙人		Development Principles Fund II GP Ltd.	
主营业务		投资	

(a) In this table, the last column is right-aligned. However, the bottom part gives all the partners’ name and their capital subscriptions. The first 8 rows are thus far apart.

盈利能力指标	2015 年	2015 年备考	变动
营业收入	565,457.86	853,949.30	288,491.44
营业利润	75,589.69	77,115.31	1,525.62
利润总额	75,949.63	77,553.66	1,604.03
净利润	63,381.65	62,410.45	-971.20
主营业务毛利率	31.21%	30.36%	-0.85%
总资产报酬率	6.72%	5.19%	-1.53%

(b) In this profitability table, the first column is left-aligned for legibility and the other three columns are right-aligned because only then can the decimal points be aligned.

Fig. 6. Typical examples of unevenly distributed horizontal cells in financial tables.

implies that our algorithm works well as a baseline given enough training data. In addition, GFTE also achieves good results on FinTab test dataset, which suggests that GCN model also works well on more complex scenario.

Table 7. Accuracy results of both vertical and horizontal relations on validation dataset and test dataset.

Dataset	Horizontal prediction	Vertical prediction
SciTSR test dataset	0.954048	0.922423
FinTab test dataset	0.883612	0.918561

In conclusion, applying Graph Convolutional Network to a table extraction problem by integrating image feature, position feature and textual feature together is a novel solution. Since tables in financial context are much more difficult than ordinary tables to be exacted by existing methods, GFTE shows that integrating more types of table feature helps to improve the performance and is thus introduced and suggested as baseline method, which is hoped to be enlightening.

6 Conclusion

In this paper, we disclose a standard Chinese financial dataset from PDF files for table extraction benchmark test, which is diverse, sufficient and comprehensive. With this novel dataset, we hope more innovative and fine-designed algorithms of table extraction will emerge. Meanwhile, we propose a GCN-based algorithm GFTE as a baseline with a novel idea of integrating all possible types of ground truth together. We also discuss its performance and some possible difficulties by extracting tables from financial files in Chinese.

References

1. Etzioni, O., Fader, A., et al.: Open information extraction: the second generation. Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence, AAAI Press, Spain, 2011, pp. 3–10 (2011)
2. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA (1999)
3. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press New York, NY, USA (2010)
4. Hurst, M.: Towards a theory of tables. *Int. J. Document Anal. Recogn. (IJ DAR)* 8(2–3), 123–131 (2006)
5. Long, V., Dale, R., Cassidy, S.: A model for detecting and merging vertically spanned table cells in plain text documents. In: International Conference on Document Analysis and Recognition, New York (2005)
6. Shigarov, A., Altaev, A., et al.: TabbyPDF: web-based system for PDF table extraction. In: 24th International Conference on Information and Software Technologies (2018)
7. Institute of Computer Science and Technology of Peking University, Institute of Digital Publishing of Founder R&D Center, “Marmot Dataset,” China (2011)
8. Shahab, A.: Table Ground Truth for the UW3 and UNLV datasets. In: German Research Center for Artificial Intelligence (DFKI) (2013)
9. Göbel, M., Hassan, T., Oro, E., Orsi, G.: ICDAR 2013 table competition. In: Proceedings of the 2013 12th International Conference on Document Analysis and Recognition, pp. 1449–1453 (2013)
10. Gao, L., Huang, Y., Dejean, H., Meunier, J.: ICDAR 2019 Competition on Table Detection and Recognition (cTDaR). In: International Conference on Document Analysis and Recognition (ICDAR), pp. 1510–1515 (2019)
11. Zhong, X., Tang, J., Yepes, A.J.: PubLayNet: largest dataset ever for document layout analysis. In: International Conference on Document Analysis and Recognition (ICDAR) (2019)
12. Chi, Z., Huang, H., Xu, H., Yu, H., Yin, W., Mao, X.: Complicated table structure recognition (2019). arXiv preprint [arXiv:1908.04729](https://arxiv.org/abs/1908.04729)
13. Li, M., Cui, L., Huang, S., Wei, F., Zhou, M., Li, Z.: TableBank: table benchmark for image-based table detection and recognition. In: The International Conference on Language Resources and Evaluation (2020)
14. Göbel, M., Hassan, T., Oro, E., Orsi, G., Rastan, R.: Table modelling, extraction and processing. In: Proceedings of the 2016 ACM Symposium on Document Engineering, pp. 1–2 (2016)
15. Coiasnon, B., Lemaitre, A.: Handbook of Document Image Processing and Recognition. Chap. Recognition of Tables and Forms, pp. 647–677 (2014)

16. Khusro, S., Latif, A., Ullah, I.: On methods and tools of table detection, extraction and annotation in PDF documents. *J. Inf. Sci.* **41**(1), 41–57 (2015)
17. Wang, Y.: Document analysis: table structure understanding and zone content classification. PhD thesis, Washington University (2002)
18. Shamilian, J.H., Baird, H.S., Wood, T.L.: A retargetable table reader. In: Proceedings of the Fourth International Conference on Document Analysis and Recognition. IEEE, New York (1997)
19. Mohemad, R., Hamdan, A.R., Othman, Z.A., Noor, N.M.: Automatic document structure analysis of structured PDF files. *Int. J. New Comput. Architect. Appl. (IJNCAA)* **1**(2), 404–411 (2011)
20. Yildiz, B., Kaiser, K., Miksch, S.: pdf2table: a method to extract table information from PDF files. In: IICAI (2005)
21. Liu, Y., Mitra, P., Giles, C.L., Bai, K.: Automatic extraction of table metadata from digital documents. In: Proceedings of the 6th ACM/IEEE-CS Joint Conference on Digital Libraries. ACM, New York (2006)
22. Wang, Y., Hu, J.: Detecting tables in HTML documents. In: Lopresti, D., Hu, J., Kashi, R. (eds.) DAS 2002. LNCS, vol. 2423, pp. 249–260. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-45869-7_29
23. Cohen, W.W., Hurst, M., Jensen, L.S.: A flexible learning system for wrapping tables and lists in HTML documents. In: Proceedings of the 11th International Conference on World Wide Web. ACM, New York (2002)
24. Oro, E., Ruffolo, M.: Xonto: an ontology-based system for semantic information extraction from pdf documents. In: ICTAI 2008: 20th IEEE International Conference on Tools with Artificial Intelligence. IEEE, New York (2008)
25. Wang, Y., Hu, J.: A machine learning based approach for table detection on the web. In: Proceedings of the 11th International Conference on World Wide Web. ACM, New York (2002)
26. Silva, A.C.: New metrics for evaluating performance in document analysis tasks - application to the table case. ICDAR (2007)
27. Liu, Y., Mitra, P., Giles, C.L.: Identifying table boundaries in digital documents via sparse line detection. In: Proceedings of the 17th ACM Conference on Information and Knowledge Management. ACM, New York (2008)
28. Pinto, D., McCallum, A., Wei, X., Croft, W.B.: Table extraction using conditional random fields. In: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion retrieval. ACM, New York (2003)
29. Wei, X., Croft, B., McCallum, A.: Table extraction for answer retrieval. *Inf. Retrieval* **9**(5), 589–611 (2006)
30. 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 (2019)
31. Pau, R., Anjan, D., Lutz, G., Alicia, F., Oriol, R., Josep, L.: Table detection in invoice documents by graph neural networks. In: 2019, International Conference on Document Analysis and Recognition (ICDAR), pp. 122–127 (2019)
32. Holeček, M., Hoskovec, A., Baudiš, P., Klinger, P.: Table understanding in structured documents. In: International Conference on Document Analysis and Recognition Workshops (ICDARW), pp. 158–164 (2019)
33. Chris, T., Morariu, V., Price, B., Cohen, S., Tony, M.: “Deep splitting and merging for table structure decomposition. In: 2019, International Conference on Document Analysis and Recognition (ICDAR), pp. 114–121 (2019)