Cracking Tabular Presentation Diversity for Automatic Cross-Checking over Numerical Facts

Hongwei Li[1,2], Qingping Yang[1,2], Yixuan Cao[1,2], Jiaquan Yao[3], Ping Luo[1,2*]
[1]Key Lab of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology, CAS, Beijing 100190, China.
[2]University of Chinese Academy of Sciences, Beijing 100049, China.
[3]School of Management, Jinan University, Guangzhou 510632, China.

ABSTRACT

Tabular forms of numerical facts widely exist in the disclosure documents of vertical domains, especially the financial fields. It is also quite common that the same fact might be mentioned multiple times in different tables with diverse *tabular presentation*. Firm's disclosure documents are the main source of accounting information for individual investors. Its authenticity is crucial for both firms' development and investors' investment decisions. However, due to large volumes of tables, frequent updates during editing, and limited time for manual cross-checking, these facts might be inconsistent with each other even after official publishing. Such errors may bring about huge reputational risk, and even economic losses even if the mistakes are made unintentionally instead of deliberately. Hence, it creates an opportunity for Automatic Numerical Cross-Checking over Tables. This paper introduces the key module of such a system, which aims to identify whether a pair of table cells are *semantically equivalent*, namely referring to the same fact. We observed that due to tabular presentation diversity the facts in tabular forms are difficult to be parsed into relational tuples. Thus, we present an end-to-end solution of binary classification over each pair of table cells, which does not involve with explicit semantic parsing over tables. Also, we discuss the design of this neural model to compromise between prediction accuracy and inference time for a large number of table cell pairs, and propose some practical techniques to address the issue of extreme classification imbalance among pairs. Experiments show that our model achieves macro $F_1 = 0.8297$ in linking semantically equivalent table cells from the IPO prospectus. Finally, an auditing tool is built to support guided cross-checking over financial documents, reducing work hours by $52\% \sim 68\%$. This system has received wide recognition in the Chinese financial community. Nine of the top ten Chinese security brokers have adopted this system to support their business of investment banking.

[*] Corresponding author: luop@ict.ac.cn

KDD '20, August 23–27, 2020, Virtual Event, USA
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-7998-4/20/08...$15.00
https://doi.org/10.1145/3394486.3403310

CCS CONCEPTS

• **Social and professional topics** → **Automation**; • **Information systems** → *Business intelligence*.

KEYWORDS

tabular presentation, numerical facts, automatic cross-checking

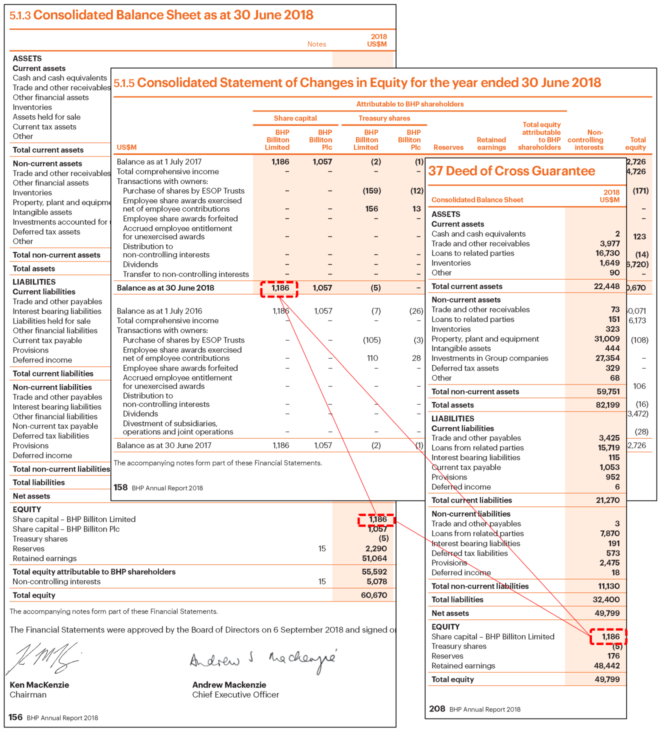
ACM Reference Format:
Hongwei Li[1,2], Qingping Yang[1,2], Yixuan Cao[1,2], Jiaquan Yao[3], Ping Luo[1,2]. 2020. Cracking Tabular Presentation Diversity for Automatic Cross-Checking over Numerical Facts. In *26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '20), August 23–27, 2020, Virtual Event, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394486.3403310

1 INTRODUCTION

Tabular forms of numerical facts widely exist in the disclosure documents of vertical domains. For example, various financial documents, e.g. IPO prospectuses, bond prospectuses, corporate annual reports etc., contain a large number of tables over the finance indicators of the corporation. Figure 1 shows the screenshot of three tables in BHP Annual Report 2018. The finance indicators in the tables are organized in a clear form with rows and columns, which enable readers easily make comparisons and better understand the firm's financial situation. In these tables, each numerical value refers to a *numerical fact* about a specific finance indicator at certain time for a given company. Based on the disclosure documents of 270 IPO prospectuses, on average there are 224 tables and 5,821 numerical fact mentions in each report. For a collection of 762 auditor's reports, these two average numbers are 65 and 1,061 for each report. The tabular forms of numerical facts provide a neat format to quantitatively present the numerical aspects of their objective indicators.

It is also quite common that the same fact might be mentioned multiple times in different tables. In Figure 1, the three table cells in red dash box are shown as an example. Although they are in different table cells, they all refer to the same fact that *the share capital for BHP Billiton Limited at the end of fiscal year 2018 is 1,186M US$*[1]. In other words, this fact is mentioned three times in three different table cells. In addition, Figure 2 shows the distribution of the table cells according to the number of times their corresponding facts mentioned in a document. Note that Figure 2 only shows the distribution of table cells in which numerical facts are mentioned more than once in a document. In total, the proportions of table

[1] Different colors in the sentence indicate different key ingredients of the fact.



5.1.3 Consolidated Balance Sheet as at 30 June 2018

	Notes	2018 US$M
ASSETS		
Current assets		
Cash and cash equivalents		
Trade and other receivables		
Other financial assets		
Inventories		
Assets held for sale		
Current tax assets		
Other		
Total current assets		
Non-current assets		
Trade and other receivables		
Other financial assets		
Inventories		
Property, plant and equipme		
Intangible assets		
Investments accounted for		
Deferred tax assets		
Other		
Total non-current assets		
Total assets		
LIABILITIES		
Current liabilities		
Trade and other payables		
Interest bearing liabilities		
Liabilities held for sale		
Other financial liabilities		
Current tax payable		
Provisions		
Deferred income		
Total current liabilities		
Non-current liabilities		
Trade and other payables		
Interest bearing liabilities		
Other financial liabilities		
Non-current tax payable		
Deferred tax liabilities		
Provisions		
Deferred income		
Total non-current liabilities		
Total liabilities		
Net assets		
EQUITY		
Share capital – BHP Billiton Limited		1,186
Share capital – BHP Billiton Plc		1,057
Treasury shares		(5)
Reserves	15	2,290
Retained earnings		51,064
Total equity attributable to BHP shareholders		55,592
Non-controlling interests	15	5,078
Total equity		60,670

The accompanying notes form part of these Financial Statements.

The Financial Statements were approved by the Board of Directors on 6 September 2018 and signed o

Andrew S Mackenzie

Ken MacKenzie
Chairman

Andrew Mackenzie
Chief Executive Officer

156 BHP Annual Report 2018

5.1.5 Consolidated Statement of Changes in Equity for the year ended 30 June 2018

	Attributable to BHP shareholders								
	Share capital		Treasury shares						
US$M	BHP Billiton Limited	BHP Billiton Plc	BHP Billiton Limited	BHP Billiton Plc	Reserves	Retained earnings	Total equity attributable to BHP shareholders	Non-controlling interests	Total equity
Balance as at 1 July 2017	1,186	1,057	(2)	(1)					2,726
Total comprehensive income	–	–	–	–					4,726
Transactions with owners:									
Purchase of shares by ESOP Trusts	–	–	(159)	(12)					(171)
Employee share awards exercised net of employee contributions	–	–	156	13					–
Employee share awards forfeited	–	–	–	–					–
Accrued employee entitlement for unexercised awards	–	–	–	–					123
Distribution to non-controlling interests	–	–	–	–					(14)
Dividends	–	–	–	–					6,720)
Transfer to non-controlling interests	–	–	–	–					–
Balance as at 30 June 2018	**1,186**	**1,057**	**(5)**	–					0,670
Balance as at 1 July 2016	1,186	1,057	(7)	(26)					0,071
Total comprehensive income	–	–	–	–					6,173
Transactions with owners:									
Purchase of shares by ESOP Trusts	–	–	(105)	(3)					(108)
Employee share awards exercised net of employee contributions	–	–	110	28					–
Employee share awards forfeited	–	–	–	–					–
Accrued employee entitlement for unexercised awards	–	–	–	–					106
Distribution to non-controlling interests	–	–	–	–					(16)
Dividends	–	–	–	–					3,472)
Divestment of subsidiaries, operations and joint operations	–	–	–	–					(28)
Balance as at 30 June 2017	1,186	1,057	(2)	(1)					2,726

The accompanying notes form part of these Financial Statements.

158 BHP Annual Report 2018

37 Deed of Cross Guarantee

Consolidated Balance Sheet	2018 US$M
ASSETS	
Current assets	
Cash and cash equivalents	2
Trade and other receivables	3,977
Loans to related parties	16,730
Inventories	1,649
Other	90
Total current assets	22,448
Non-current assets	
Trade and other receivables	73
Loans to related parties	151
Inventories	323
Property, plant and equipment	31,009
Intangible assets	444
Investments in Group companies	27,354
Deferred tax assets	329
Other	68
Total non-current assets	59,751
Total assets	82,199
LIABILITIES	
Current liabilities	
Trade and other payables	3,425
Loans from related parties	15,719
Interest bearing liabilities	115
Current tax payable	1,053
Provisions	952
Deferred income	6
Total current liabilities	21,270
Non-current liabilities	
Trade and other payables	3
Loans from related parties	7,870
Interest bearing liabilities	191
Deferred tax liabilities	573
Provisions	2,475
Deferred income	18
Total non-current liabilities	11,130
Total liabilities	32,400
Net assets	49,799
EQUITY	
Share capital – BHP Billiton Limited	1,186
Treasury shares	(5)
Reserves	176
Retained earnings	48,442
Total equity	49,799

208 BHP Annual Report 2018

Figure 1: The screenshot of three tables in BHP Annual Report 2018. These three tables are deliberately overlapped in order to save space. The lower-left corner of each table shows its page number in the original report. The three cells in red dash box mention the same numerical fact in this report.

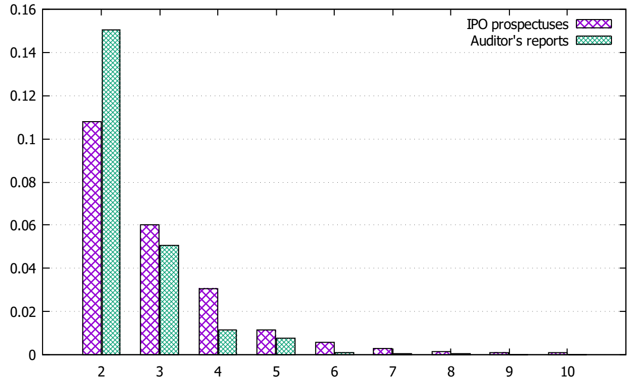


Figure 2: The distribution of numerical table cells according to the number of times its corresponding fact is mentioned in a document.

cells with multiple fact mentions amount to 22.43% and 22.24% in IPO prospectuses and auditor's reports, respectively.

Due to large volumes of tables, significant fraction of table cells have multiple fact mentions, and tables frequently update during collaborative editing, these numerical facts might be inconsistent among their multiple mentions. For example, if one of the three table cells in red dash box of Figure 1 changes to a number *not equal* to "1,186", the inconsistency among these fact mentions occurs. Such errors can seriously affect readers' assessment of the company and may cause them to doubt the reliability of the whole process and undervalue the firm even if the mistakes are made unintentionally instead of deliberately. Some recent news reported that these numerical errors brought about huge reputation risk, and even economic losses [1]. Since the documents disclosed by the firm usually have the force of law, these errors should be thoroughly removed before officially publishing.

Extant studies have found that the negative effect of errors cannot be ignored and is more severe than management's anticipation. Lawrence [8] points out that investors are more willing to invest firms with clear and concise financial information, so accounting errors also deserve our attention. Choudhary, Merkley, and Schipper [2] find that investors believe that even immaterial errors mean weak corporate governance or poor quality of financial reports. Fang, Huang, and Wang [4] reveal that errors would affect the investors' reactions to firm's earnings surprises and abilities to detect fraud. Overall, investors' attention to the accounting errors can lead to damage to firms' reputation.

Traditionally, there is a special job called *authorized reading* to manually conduct numerical cross-checking. Based on the interview to the employees from one of the worldwide top 4 accounting firms, it takes an experienced professional 1 hour for the task of cross-checking over 10 pages. Additionally, there is usually a hard deadline, e.g. April 30 for the disclosure of annual reports of listed company in China, to publish the disclose documents, and the time left for cross-checking is usually limited. More importantly, repeated reading back and forth definitely induces fatigue, tiredness, and carelessness. Hence, these data-inconsistency errors are still inevitable even after manual cross-checking.

Therefore, it creates an opportunity for automated numerical cross-checking systems. There are some related systems developed, such as ClaimBuster [5] and StatCheck [12]. ClaimBuster focused on detecting check-worthy factual claims while the other two components of matching claims and checking claims are still ongoing. StatCheck uses rule-based program to check inconsistency errors in the null-hypothesis significance testing, presented in the academic papers in major psychology journals. A recent study [1] published a system called AutoDoc, and introduced the module of cross-checking among only *textual paragraphs*. Since tables are more efficient to organize and summarize data, there are much more numerical facts in tables than textual paragraphs. Therefore, as an important extension to [1], we propose Automatic Numerical Cross-Checking over Tables (ANCOT) in this study.

The key module of such a system is to identify whether a pair of two table cells are *semantically equivalent*, namely referring to the same fact. With the support of this module, it is easy to automatically identify inconsistent errors that the numerical values inside two semantically equivalent table cells are not equal. However, although table provides a semi-structured form to organize data it also provides much freedom to place the key ingredients of a fact into different table areas, namely row header, column header, or even the context outside table. Due to this *tabular presentation diversity* it is not trivial to parse the key ingredients of the fact in each table cell. To address this issue, we present an end-to-end solution of binary classification to judge whether any pair of table cells is semantically equivalent or not without explicit semantic parsing. Also, we propose some practical techniques to address the issues of huge number of table cell pairs and extreme classification imbalance among them.

Experiments show that our classification model achieves macro F_1 0.7944 and 0.8297 in auditor's reports and IPO prospectuses, respectively. We also show some cases that the model can deal with the tabular presentation diversity. Meanwhile, the time in handling a document with hundreds of pages is within a few minutes. This time consumption is acceptable in practical scenarios.

Finally, we have built an auditing tool to support cross-checking over numerical tables in financial documents, such as IPO Prospectuses and auditor's reports. This system provides a method for identifying the numerical inconsistency errors in the tables of financial documents, which is conducive to improving the efficiency of auditing work. In practice, it reduces about 50% work hours in checking IPO prospectuses and auditor's reports. Currently, this system has received wide recognition in the Chinese financial community. Nine of the top ten Chinese security brokers have adopted this system to support their business of investment banking.

2 PROBLEM FORMULATION WITH NOTIONS AND DENOTATIONS

In this section, we first give some notions and denotations, then discuss the issue of tabular presentation diversity, and finally formulate the problem.

2.1 Notions and Denotations

Although *entity* and *relational* tables are prevalent on the Web [9], *matrix tables* are more important in vertical domains since they have more concise layout and are easier to understand by human than relational tables. For the financial disclosure documents, the proportion of matrix tables is as high as 90% based on our empirical study. Thus, this study mainly considers the numerical facts in matrix tables.

Figure 3(a) shows a typical matrix table, and Figure 3(b) illustrates that a matrix table usually consists of 4 table areas, namely *column headers*, *row headers*, *data cells* and *context*. Here, the *column headers*, *row headers* and *data cells* are the areas inside a table, while the *context* is outside a table, including the descriptive text in the title or subtitle of a table. In this study, we assume that all these table areas are identified in some preprocessed steps.

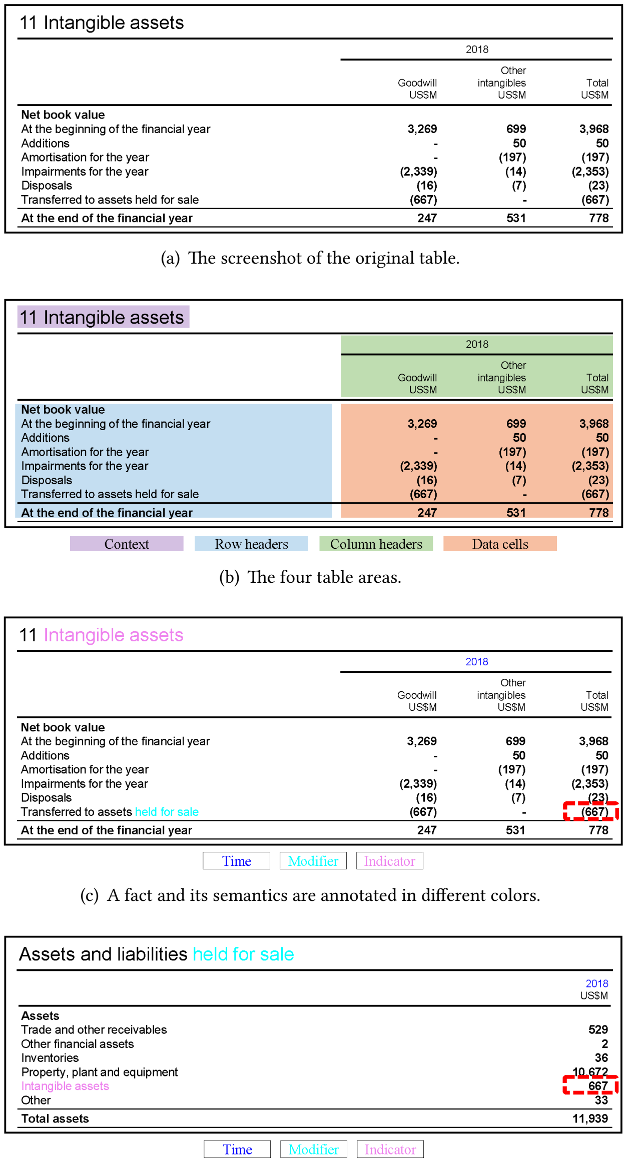
Definition 2.1. **Fact mention.** Each table cell in the area of data cells refers to a *fact mention*. In our application, the semantics of each fact mention can be described as a triple $\mathbf{f}$:

$$(\mathit{time}, \mathit{modifier}, \mathit{indicator}),$$

where *time* is the time for this fact, *indicator* is the financial indicator this fact refers to, and *modifier* is the modifier of the financial indicator.

For example, the cell in the red box of Figure 3(c) refers to a fact mention as follows,

$$(2018, \mathit{held\ for\ sale}, \mathit{intangible\ assets}),$$



11 Intangible assets

	2018		
	Goodwill US$M	Other intangibles US$M	Total US$M
Net book value			
At the beginning of the financial year	3,269	699	3,968
Additions	-	50	50
Amortisation for the year	-	(197)	(197)
Impairments for the year	(2,339)	(14)	(2,353)
Disposals	(16)	(7)	(23)
Transferred to assets held for sale	(667)	-	(667)
At the end of the financial year	247	531	778

(a) The screenshot of the original table.

11 Intangible assets

	2018		
	Goodwill US$M	Other intangibles US$M	Total US$M
Net book value			
At the beginning of the financial year	3,269	699	3,968
Additions	-	50	50
Amortisation for the year	-	(197)	(197)
Impairments for the year	(2,339)	(14)	(2,353)
Disposals	(16)	(7)	(23)
Transferred to assets held for sale	(667)	-	(667)
At the end of the financial year	247	531	778

(b) The four table areas.

11 Intangible assets

	2018		
	Goodwill US$M	Other intangibles US$M	Total US$M
Net book value			
At the beginning of the financial year	3,269	699	3,968
Additions	-	50	50
Amortisation for the year	-	(197)	(197)
Impairments for the year	(2,339)	(14)	(2,353)
Disposals	(16)	(7)	(23)
Transferred to assets held for sale	(667)	-	(667)
At the end of the financial year	247	531	778

(c) A fact and its semantics are annotated in different colors.

Assets and liabilities held for sale

	2018 US$M
Assets	
Trade and other receivables	529
Other financial assets	2
Inventories	36
Property, plant and equipment	10,672
Intangible assets	667
Other	33
Total assets	11,939

(d) Another equivalent fact with different tabular presentation.

Figure 3: The example of tabular presentation diversity. The two tables in Figures 3(a) and 3(d) are in Pages 180 and 199 of BHP Annual Report 2018, respectively.

Definition 2.2. **Semantically equivalent of two fact mentions.** Given two fact mentions $\mathbf{f}_1$ and $\mathbf{f}_2$ as follows,

$$(\mathit{time}_1, \mathit{modifier}_1, \mathit{indicator}_1)$$

$$(\mathit{time}_2, \mathit{modifier}_2, \mathit{indicator}_2)$$

we say they are semantically equivalent if and only if time_1 and time_2 refer to the same time, $\mathit{modifier}_1$ and $\mathit{modifier}_2$ are semantically matched, $\mathit{indicator}_1$ and $\mathit{indicator}_2$ refer to the same financial indicator.

Since each numerical cell corresponds to its fact mention, we call two numerical cells are *semantically equivalent* if and only if their corresponding two fact mentions are semantically equivalent. It is clear that the cell in the red box of Figure 3(c) is semantically equivalent to the one in Figure 3(d).

2.2 Problem Formulation and Analysis

Problem Formulation. Given all the numerical cells from all the tables of a document, we identify all the pairs of numerical cells, each of which are semantically equivalent.

Problem Analysis. Once we obtain the result to this problem, it is easy to check whether the numerical values inside two semantically equivalent table cells are equal or not. Thus, such inconsistency errors can be automatically detected. However, in this problem, we need to address the following challenges.

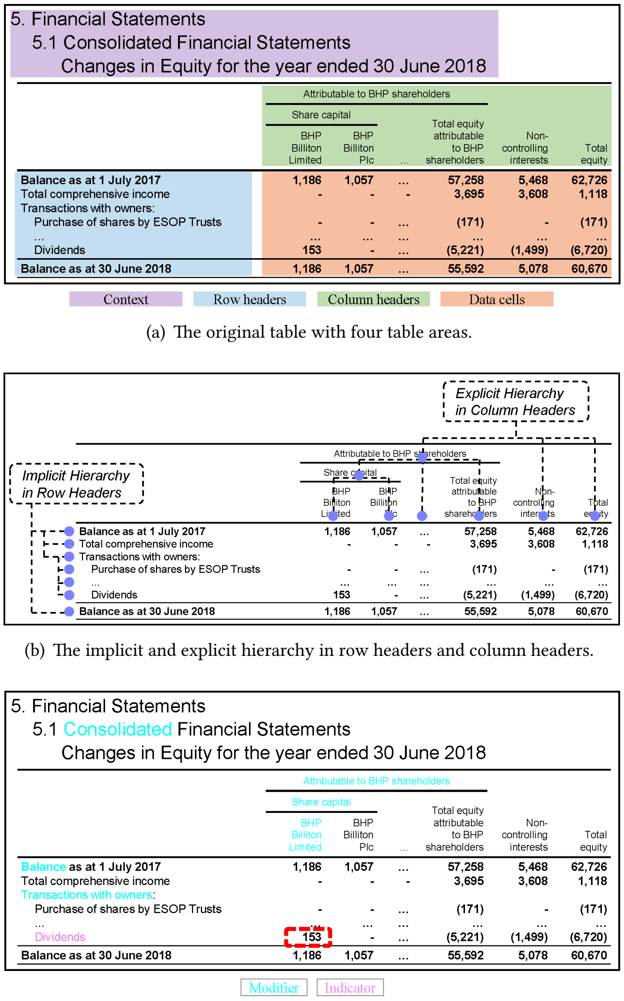
- *Tabular Presentation Diversity.* Although tabular form is semi-structured, it also provides much freedom to place the key ingredients of a fact into different table areas. It indicates that the *time*, *modifier*, and *indicator* of a fact can be scattered at everywhere in table areas. For example, as shown in Figure 3(c), the *time*, *modifier*, and *indicator* of the fact in the red dashed box are located in the column header, row header, and context, respectively. However, as shown in Figure 3(d), these three locations are the column header, context, and row header. Only the location of *time* remains the same, while the locations of $modifier$ and $indicator$ change. The situation becomes more complicated when the table contains hierarchical headers (detailed in Section 3.1). Due to the tabular presentation diversity, it is not trivial to explicitly parse the key ingredients of the fact in each table cell.
- *Huge Number of Table Cell Pairs.* As mentioned earlier, on average there are thousands of numerical cells in a financial document, resulting in millions of table cell pairs. Thus, we need to consider the compromise between prediction accuracy and inference time for such a huge number of instances.
- *Extreme Classification Imbalance.* For such a huge number of table cell pairs, only a tiny fraction of pairs are semantically equivalent. Based on our disclosure documents, on average the positive to negative ratios are smaller than 1:12,000 and 1:6,000 for IPO prospectuses and auditor's reports, respectively. We need to carefully consider this issue to guarantee high accuracy.

3 SOLUTIONS

We first describe our end-to-end model to determine whether a pair of numerical cells are semantically equivalent or not. Then, we introduce the grouping and deduplication methods to reduce the number of pairs that need to be classified.

3.1 Cell Pair Classification

In our solution, we do not explicitly extract the fact mention of numerical cells for classification. Instead, we propose an end-to-end model that directly predicts whether two cells refer to the same fact mention. Our model consists of two parts: cell embedding network and pair classification network. The cell embedding network takes as input the numerical cell and the table that it is located in, and outputs a representation for the cell. The pair classification network takes as input two representations of two cells and predict whether they refer to the same fact.


5. Financial Statements
5.1 Consolidated Financial Statements
Changes in Equity for the year ended 30 June 2018

	Attributable to BHP shareholders					
	Share capital					
	BHP Billiton Limited	BHP Billiton Plc	...	Total equity attributable to BHP shareholders	Non-controlling interests	Total equity
Balance as at 1 July 2017	1,186	1,057	...	57,258	5,468	62,726
Total comprehensive income	-	-	-	3,695	3,608	1,118
Transactions with owners:						
Purchase of shares by ESOP Trusts	-	-	...	(171)	-	(171)
...	...	...	...	...	...	...
Dividends	153	-	...	(5,221)	(1,499)	(6,720)
Balance as at 30 June 2018	1,186	1,057	...	55,592	5,078	60,670

(a) The original table with four table areas.

Explicit Hierarchy in Column Headers
Implicit Hierarchy in Row Headers

(b) The implicit and explicit hierarchy in row headers and column headers.

(c) A fact and its semantics are annotated in different colors.

Figure 4: Another example of tabular presentation diversity. The table in Figures 4(a) is in Pages 158 BHP of Annual Report 2018.

Cell Embedding Network. Given a cell and the table that it is located in, this network embeds the fact mention of this cell into a dense representation. The first question is what we should feed into the network. Feeding only the numerical value of the cell contains no information of the fact mention. Feeding the entire table is unnecessary, and dilutes the information of the specific cell. As we are handling matrix table, a straightforward idea would be feeding the corresponding row and column header of the cell. But they may not contain enough information.

Figure 4(a) shows a typical table from real world documents. There are hierarchical structures in row and column headers, as shown in Figure 4(b). The hierarchy is usually in the form of a tree structure. In Figure 4(b), the column headers have an explicit three levels hierarchy, presented using different rows and merged cells. The row headers have three levels implicit hierarchy, presented by their visual cures (e.g. font styles, indentation). Such hierarchy is frequently used to reduce redundant expressions inside a table for human reading, but poses a challenge for machine understanding.

We take the cell in the red dotted box in Figure 4(c) as an example to show how to include all the information of its fact mention. The ingredients of its fact mention are highlighted. In row headers, there are (Balance as at 1 July 2017, Transactions with owners:, Dividend). In column headers, there are (Attributable to BHP shareholders, Share capital, BHP Billiton Limited). As "Dividend" and "BHP Billiton Limited" is the corresponding row and column header of the cell, these ingredients lies right on the path from root to the corresponding header. The situation in context is similar. Since *time* information in fact mention has less diversity in format, we omit time information in our model, and process it by rule.

Therefore, we feed the root to leaf path of the row header, column header and context into the model. Specifically, we define $R = (r_1, \ldots, r_l)$ as the row header input, where l is the number of nodes from root to leaf, and $r_1 = (w_1, ..., w_n)$ is the text in the root cell. In the example, R=(Balance as at 1 July 2017, Transactions with owners:, Dividend), and r_1=(Balance, as, at, 1, July, 2017). Similarly, we define C as the column header input, and we distinguish table title and section titles in context as T and S. In summary, the input of cell embedding network of a cell is (R, C, T, S).

The second question is what is the form of its representation and how to compute it. As we can see from Figure 3, different areas of a table might convey different kinds of information, so we give one hidden vector for each component of (R, C, T, S) as the representation of the numerical cell, namely (h_R, h_C, h_T, h_S). We encode R by using two LSTMs. The first LSTM takes as input a text r_i and returns a vector h_{r_i}. The second LSTM takes as input $(h_{r_1}, \cdots, h_{r_l})$ and outputs a vector h_R:

$$
\begin{aligned}
h_{r_i} &= \mathrm{LSTM}_1(r_i), \qquad \text{for } r_i \in R \\
h_R &= \mathrm{LSTM}_2(h_{r_i}, \ldots, h_{r_l})
\end{aligned}
$$

Similarly, C, T and S share the same encoding network to compute their hidden vectors. Finally, cell embedding network outputs the representation of a numerical cell as a tuple of vectors $H = (h_R, h_C, h_T, h_S)$.

Pair Classification Network. The pair classification network takes as inputs two representations H and H' of numerical cells c and c', and outputs the probability that they are semantically equivalent.

First, each part of H, namely h , attends on H' to compose a new vector h_a that lays emphasis on certain counterparts in H':

$$
\begin{aligned}
h_a &= \mathrm{Attn}(h, H') \text{ for } h \in H, \\
H_a &= (h_{Ra}, h_{Ca}, h_{T_a}, h_{Sa})
\end{aligned}
$$

Here, the Attn module is composed of a dot scaled attention layer and Feed Forward Network (FFN) layer, each followed by a Residual Connection (RC) and Layer Normalization (LN):

$$
\begin{aligned}
Z_1 &= \mathrm{Attention}(Q, K, V) = \mathrm{softmax}(\frac{Q^T K}{\sqrt{d_h}} V) \\
Z_2 &= \mathrm{LN}_1(Q + Z_1) \\
Z_3 &= \mathrm{FFN}(Z_2) \\
Q' &= \mathrm{LN}_2(Z_2 + Z_3)
\end{aligned}
$$

where $Q = h$, $K = H'$, and $V = H'$. Similarly, each part h' of H' attends on H to get H'_a.

Then we want one vector representation of each cell, so we use the other Attn module, and use a special learnable vector e to get two vectors: $u = \mathrm{Attn}(e, H_a)$, $v = \mathrm{Attn}(e, H'_a)$.

Finally, we predict the probability of semantically equivalence. In order to ensure the symmetry of the result (to output similar result if we swap the order of two cells), u and v are concatenated in two different orders, and the probability is computed by:

$$
\begin{aligned}
s_1 &= \mathrm{FFN}([u; v]) \\
s_2 &= \mathrm{FFN}([v; u]) \\
p &= \mathrm{softmax}(\max(s_1, s_2))
\end{aligned}
$$

where $\max(\cdot)$ indicates an element-wise max function.

3.2 Filtering

As we have analyzed in Section 2, there are millions of pairs to be classified, so we proposed two methods to reduce the number of pairs. The first step is to quickly filter out pairs of numerical cells, each of which are not semantically equivalent. The second step is to remove highly similar pairs to reduce duplicate calculations. Next, we describe these two steps: grouping and deduplication.

Grouping. Although the tabular presentation is diverse, it is not difficult to parse the *time* and *numeric type* for each table cell. The time of a fact mention is usually located in its corresponding table headers or context. Also, we consider two numeric types, *proportion* (between 0 and 1) and *other amount*. All these entities can be recognized by the regular expressions. If more than one occurrences of time are extracted, the one inside the table takes precedence. Hence, for each cell there are a *tag* of time and numeric type on it.

It is clear that the two table cells with different time or numeric type cannot be semantically equivalent. Thus, we can group the table cells with their tags, and only classify the pairs of numerical cells within each group.

Deduplication. Disclosure documents often describe the financial indicators of a company for recent years (usually 2 - 4 years). Thus, there are some numerical cells whose fact mentions are different in terms of *time* while their $modifier$ and $indicator$ are exactly the same. In such a situation, we call the two numerical cells c and c' are *highly similar*, denoted as $c \approx c'$

For example, two tables are shown in Figure 5. In each table, the cells in red and blue dashed boxes are highly similar, since they share the same row header and context, but have different *times* in column headers.

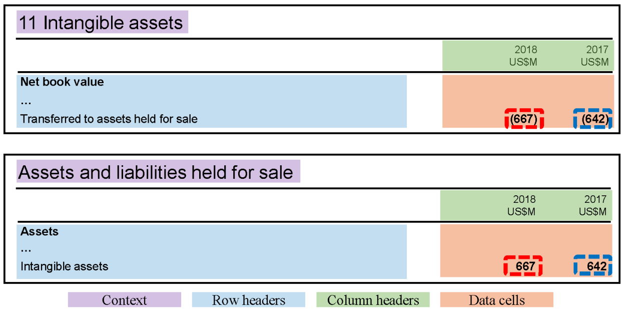


Figure 5: Two simplified tables.

Then, we define *near duplicate pairs*. Given two pairs (c, c') and (d, d'), they are near duplicate, if and only if $c \approx d$, $c' \approx d'$, c and c' refer to the same time, d and d' refer to the same time.

For example, in Figure 5 we can get the following two pairs of numerical cells: one with the cells in the two red dashed boxes, another with those in the two blue dashed boxes. It is easy to check that these two pairs are highly similar.

Since the input to cell embedding network ignores the time of a cell, the inputs of near duplicate pairs are the same, and judgements of semantic equivalence are the same. Hence, we only need to preserve one pair in near duplicate pairs both in the training and inference phase.

4 EXPERIMENT SETTINGS

4.1 Dataset

We collected two sets of Chinese financial documents: IPO prospectuses and auditor's reports. The detailed information of them are shown in Table 1. IPO prospectuses have more pages, tables and numerical cells than auditor's reports in general. On average, a table contains 26 numerical cells in IPO prospectuses and 16 numerical cells in auditor's reports.

The datasets are annotated as follows. We annotate each document by two annotators. The first one annotates, and then the second one proof-reads the results from the first annotator. For a fact mentioned in n numerical cells, there are $\binom{n}{2}$ semantically equivalent pairs among them. To reduce the annotation efforts, we require that a numerical cell only needs to be paired with the *nearest* semantically equivalent one in front of it. After manual annotation, the annotated results can be automatically expanded to get all pairs of semantically equivalent numerical cells. After annotation, we divide each dataset into training, validation and test set by documents in the ratio of 8:1:1.

Table 1: The detailed dataset statistics.

	IPO prospectuses			Auditor's reports		
#Documents	270			762		
	Max	Min	Avg	Max	Min	Avg
#Pages	854	235	429	203	9	47
#Tables	531	95	224	256	9	65
#Numerical cells	24,042	2,692	5,822	7,876	70	1,061

We give statistics on the datasets before and after filtering to show the effectiveness of our proposed filtering methods. A pair of numerical cells is considered as positive if they are semantically equivalent, otherwise it is a negative pair. Before filtering, the ratios of negative to positive samples are 12,702:1 and 6,716:1 in IPO prospectuses and auditor's reports respectively. After grouping, 82.42% and 58.1% of samples are filtered respectively, and the ratios of negative to positive samples reduce to 2,232:1 and 2,813:1 respectively.

After deduplication, 7.05% and 12.8% of more samples are filtered respectively, and the ratios of negative to positive samples go up to 3,822:1 and 3,131:1 respectively. This means that positive samples have more highly similar pairs than negative samples.

In summary, after filtering, 89.47% and 70.9% samples are filtered out in IPO Prospectuses and auditor's reports respectively, and the ratios of negative samples to positive samples reduce to 3,822:1 and 3,131:1 respectively. The step of filtering not only reduces the number of samples, but also alleviate the issue of class imbalance. In the following the training, validation and test sets used are all after filtering.

4.2 Parameter Settings

Documents in our datasets are in Chinese and each text in table cells is not too long, thus we use character-based model in cell embedding network. The vocabulary contains 2,500 most frequent characters. Word2vec [13] is adopted to initialize character embeddings. The dimensions of the character embedding and LSTM$_1$ are set to 128. The dimension of LSTM$_2$ is set to 256. In pair classification network, the dimensions of two attentions are both set to 256, and each feed forward network is a fully connected layer with two linear transformations (256×512 and 512×256) and a ReLU activation in between. We use Adam optimization method with learning rate 0.001. In our experiments, we leverage GPU (GeForce GTX 1080 Ti) to train and infer. During training, 4 GPUs are used with total batch size 4k; during inference, 1 GPU is used to infer a document with batch size of 4k for cell embedding and 40k for matching.

4.3 Evaluation Metrics

In the problem of numerical cross-checking over tables, the main task is to determine whether any two numerical cells in different tables are semantically equivalent or not. Therefore, the predicted results and the ground truth of the task are both a set of pairs of semantically equivalent numerical cells for each document. We first define precision, recall and F_1 measures in a document. Given the predicted relation set r and ground truth set r^* in a document, they are defined as

$$\begin{aligned} P &= \frac{|r \cap r^*|}{|r|} \\ R &= \frac{|r \cap r^*|}{|r^*|} \\ F_1 &= \frac{2 * P * R}{P + R} \end{aligned}$$

Now we introduce the metric on a dataset $D = (d^{(1)}, ..., d^{(n)})$, where $d^{(i)}$ indicates the i-th document, n is the number of documents. We define Macro metrics by averaging results upon documents. For example, Macro precision on D is defined as

$$Macro\ P = \frac{\sum_{i=1}^{n} P_i}{n},$$

where P_i is the precision of the i-th document, r_i and r_i^* are the predicted relation set and ground truth set of the i-th document. We define Micro metrics that merges pairs of all documents as one document. For example, Micro precision is defined as

$$Micro\ P = \frac{\sum_{i=1}^{n} \left| r_i \cap r_i^* \right|}{\sum_{i=1}^{n} |r_i|}.$$

5 EXPERIMENT RESULTS

5.1 Effectiveness

The results of two datasets on test set are shown in Table 2. Despite the tabular presentation diversity and extreme classification imbalance challenges, our model achieves a good performance: around 0.8 Macro F_1 on both datasets.

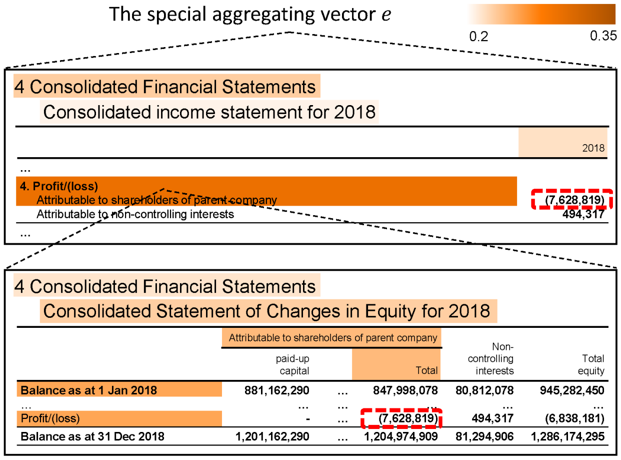
Case study on tabular presentation diversity. In cell embedding network, we give each numerical cell a representation which composes of four vectors for row headers, column headers, table title and section headings. These vectors attend on the representation of the other cells in pair classification network by two layers of attention module. Since the result of attention module can be visualized and interpreted, we use a case study to illustrate how our model can deal with the challenge of tabular presentation diversity.

In Figure 6, there are two cells in red boxes in two tables. Our model predicts that they are semantically equivalent. We visualize the weights of attention by shading, where cells with darker shading have larger attention weights. First, the table at the top of the figure shows the weights of its components when e (the special aggregating vector) attends them (H_a). It shows that the key information for this classification is the content in row headers of the cell in red.

We further display how the row headers in the table above attends on the bottom table. The colors show that the row headers and column headers in the table below are the reason why the row headers in the table above are important. This distribution is in line with the situation that the two ingredients of the row header in top table: profits, and attributed to shareholders of parent company, is located in row and column headers in the bottom table. From this case, we can see that our model design for tabular presentation diversity is effective.

Table 2: The performances of our model on two datasets.

		IPO prospectuses	Auditor's reports
Micro	P	0.8457	0.7475
	R	0.7828	0.7597
	F_1	0.8130	0.7535
Macro	P	0.8559	0.8203
	R	0.8073	0.7789
	F_1	0.8297	0.7944



4 Consolidated Financial Statements

Consolidated income statement for 2018

	2018
...	
4. Profit/(loss)	
Attributable to shareholders of parent company	(7,628,819)
Attributable to non-controlling interests	494,317
...	

4 Consolidated Financial Statements

Consolidated Statement of Changes in Equity for 2018

	Attributable to shareholders of parent company				
	paid-up capital		Total	Non-controlling interests	Total equity
Balance as at 1 Jan 2018	881,162,290	...	847,998,078	80,812,078	945,282,450
...	...	...	...	...	...
Profit/(loss)	-	...	(7,628,819)	494,317	(6,838,181)
Balance as at 31 Dec 2018	1,201,162,290	...	1,204,974,909	81,294,906	1,286,174,295

Figure 6: The illustration of the interpretability of pair classification network by a case. The cells in the red box refer to the same fact.

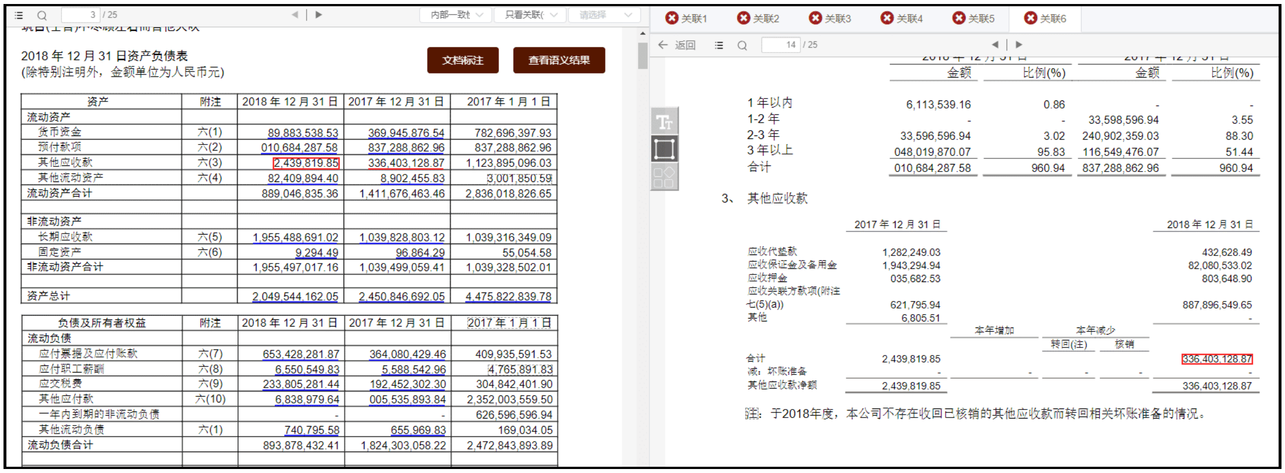
5.2 Efficiency

We evaluate the efficiency of our classification model. Since 89.47% and 70.9% pairs have been filtered out in IPO prospectuses and auditor's reports respectively, there are still on average 2,074,991 and 367,109 pairs per document in two datasets. On these two datasets, the average inference time are 428 and 74 seconds per document. In other words, on average the model can process about 4,900 pairs per second. It is acceptable in real-world scenarios if a document can be processed within a few minutes.

5.3 Real-World Application and Evaluation on Time Saving

We have built an auditing tool to support guided cross-checking over numerical facts in financial documents. To ensure that the documents after cross-checking are free of errors, we propose the following use mode of *guided cross-checking*: first the proposed model is used to identify the possible errors, and then these results are manually revised.

Figure 7 shows the screenshot of our system which supports convenient manual revision. See the left panel in this screenshot, which includes the cross-checking results from the model. Specifically, the cells without underlines indicate that their corresponding fact mentions may occur only once, while the cells with underlines indicate that these fact mentions appear more than once. Additionally, the blue underlines indicate that these fact mentions are consistent, while the red underlines indicates that some errors may exist in them. If you click a cell with red underline, a red box appears on it and at the same time the right panel jumps to the exact position of one of its equivalent table cells. This screenshot shows a true error of the same fact mention with two different values. Also, the 6 tabs on the right panel shows all the table cells which are semantically equivalent to the marked the on the left panel. With this easy interface, professional auditors can browse the document on the left panel to remove wrong pairs, and add some new pairs not recognized by our model.



2018 年 12 月 31 日资产负债表
(除特别注明外，金额单位为人民币元)

资产	附注	2018 年 12 月 31 日	2017 年 12 月 31 日	2017 年 1 月 1 日
流动资产				
货币资金	六(1)	89,883,538.53	369,945,876.54	782,696,397.93
预付款项	六(2)	010,684,287.58	837,288,862.96	837,288,862.96
其他应收款	六(3)	2,439,819.85	336,403,128.87	1,123,895,096.03
其他流动资产	六(4)	82,409,894.40	8,902,455.83	3,001,850.59
流动资产合计		889,046,835.36	1,411,676,463.46	2,836,018,826.65
非流动资产				
长期应收款	六(5)	1,955,488,691.02	1,039,828,803.12	1,039,316,349.09
固定资产	六(6)	9,294.49	96,864.29	55,054.58
非流动资产合计		1,955,497,017.16	1,039,499,059.41	1,039,328,502.01
资产总计		2,049,544,162.05	2,450,846,692.05	4,475,822,839.78

负债及所有者权益	附注	2018 年 12 月 31 日	2017 年 12 月 31 日	2017 年 1 月 1 日
流动负债				
应付票据及应付账款	六(7)	653,428,281.87	364,080,429.46	409,935,591.53
应付职工薪酬	六(8)	6,550,549.83	5,588,542.96	4,765,891.83
应交税费	六(9)	233,805,281.44	192,452,302.30	304,842,401.90
其他应付款	六(10)	6,838,979.64	005,535,893.84	2,352,003,559.50
一年内到期的非流动负债		-	-	626,596,596.94
其他流动负债	六(1)	740,795.58	655,969.83	169,034.05
流动负债合计		893,878,432.41	1,824,303,058.22	2,472,843,893.89

	2018 年 12 月 31 日		2017 年 12 月 31 日	
	金额	比例(%)	金额	比例(%)
1 年以内	6,113,539.16	0.86	-	-
1-2 年	-	-	33,598,596.94	3.55
2-3 年	33,596,596.94	3.02	240,902,359.03	88.30
3 年以上	048,019,870.07	95.83	116,549,476.07	51.44
合计	010,684,287.58	960.94	837,288,862.96	960.94

3、 其他应收款

	2017 年 12 月 31 日	本年增加	本年减少		2018 年 12 月 31 日
			转回(注)	核销	
应收代垫款	1,282,249.03				432,628.49
应收保证金及备用金	1,943,294.94				82,080,533.02
应收押金	035,682.53				803,648.90
应收关联方款项(附注七(5)(a))	621,795.94				887,896,549.65
其他	6,805.51				-
合计	2,439,819.85				336,403,128.87
减：坏账准备	-	-	-	-	-
其他应收款净额	2,439,819.85				336,403,128.87

注：于2018年度，本公司不存在收回已核销的其他应收款而转回相关坏账准备的情况。

Figure 7: The screenshot of our auditing tool that supports guided cross-checking.

Next, we compare the time of this guided cross-checking approach with the entirely manual approach. To give a more detailed analysis on how time is saved from our application, we build a mathematical model to compare these two approaches.

Let the number of numerical cells in a document be N_c, the ground-truth proportion of positive pairs be P_o. Then, the number of pairs of numerical cells in a document is $\binom{N_c}{2}$. With the Micro precision P and recall R of our model, we can obtain

$$N_{tp} = \binom{N_c}{2} * P_o * R$$
$$N_{fp} = (\frac{1}{P} - 1) * N_{tp}$$
$$N_{fn} = (\frac{1}{R} - 1) * N_{tp}$$

where N_{tp}, N_{fp}, N_{fn} are the numbers of the truth positive, false positive and false negative pairs.

Next, we assume that a professional auditor needs T_c time to determine whether an existing cell has a semantically equivalent cell, T_r time to check and correct a cell pair, and T_a time to add a new relationship between two cells. For the entirely manual approach, its time of cross-checking a document is

$$T_1 = T_c * N_c + T_a * (N_{tp} + N_{fn}),$$

while the time of our approach is

$$T_2 = T_c * N_c + T_a * N_{fn} + T_r * (N_{tp} + N_{fp})$$

Therefore, the difference is $T_1 - T_2 = T_a * N_{tp} - T_r * (N_{tp} + N_{fp})$. It clearly shows that our model saves the time of adding true positive pairs at the cost of checking the predicted positive pairs.

The interview with the professional auditors tells that with the entirely manual approach on average it takes about 18 hours to process an IPO prospectus with hundreds of pages , and 2 hours for an auditor's report with tens of pages. Based on these facts, we estimate the time required for each operation. Specifically, we calculate $T_a = (T_1 - T_c * \bar{N}_c)/(\bar{N}_{tp} + \bar{N}_{fn}) = 37s$ in the IPO prospectuses, where $\bar{N}_c = 5,822$, $\bar{N}_{tp} = 1,105$ and $\bar{N}_{fn} = 307$ that are set to their average values of our IPO prospectuses dataset. Similarly, we obtain $T_a = 25s$ in auditor's reports with $\bar{N}_c = 1,061$, $\bar{N}_{tp} = 155$ and $\bar{N}_{fn} = 49$ are set to their averages. As a result, we set T_a to [32s, 42s] and [20s, 30s] for the IPO prospectuses and auditor's reports, respectively. Additionally, T_c and T_r are set to 2s and 4s respectively, which are over-estimated to decreases the advantage of our approach since viewing a cell and a pair is quite simple.

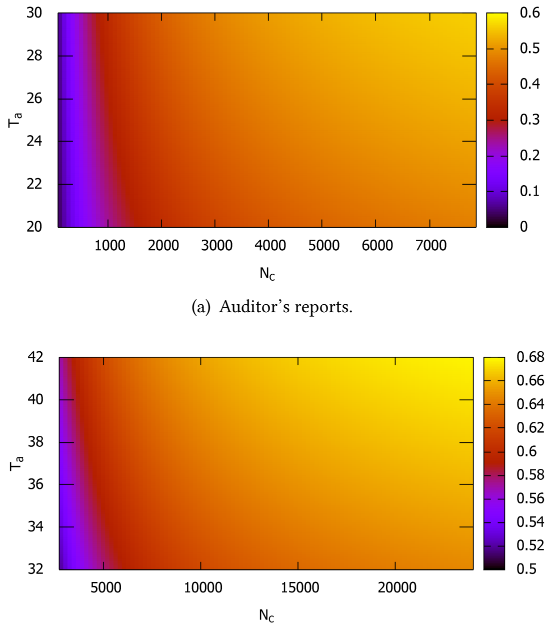
According to the assumptions above, as shown in Figure 8 we draw the proportion of saved time along the increase of T_a and N_c. The proportions of time saving are in the range of $52\% \sim 68\%$ and $5\% \sim 60\%$ for the IPO prospectuses and auditor's reports, respectively. Although this range for the auditor's reports is large, the majority is still in $40\% \sim 60\%$ when the report contain more pages.

After we deployed the system, we collected feedbacks from the actual users. They confirmed that with our system the time of guided cross-checking is shortened to 7 hours for an IPO prospectus and 1 hour for an auditor's report. These feedbacks coincide with the above theoretical evaluation on time saving.

Overall, this system has received wide recognition in the Chinese financial community. Nine of the top ten Chinese security broker adopted this system to support their business of investment banking.

6 RELATED WORK

Claim-Checking is an important issue in academic, financial, and politic fields. It has attracted a lot of research interests in recent years. Cao et al. [1] propose a system to cross-check numerical facts by extracting structured formulas from *textual paragraph* in financial documents. Our study extend their work to cross-check numerical facts in *tables*. And we adopt an end-to-end approach to avoid the extraction of explicit structured information which is laborious when collecting the labelling dataset. Vlachos and Riedel [17] propose a dataset to verify the claims made by public figures. Verifying such claims includes detecting whether a statement in check-worthy [5], retrieve information from large data source



(a) Auditor's reports.

(b) IPO prospectuses.

Figure 8: The proportion of saved time with respect to T_a and N_c.

to provide related evident paragraphs, and finally give a classification [7, 10, 16]. In academic field, Nuijten et al. [12] propose StatCheck which uses rule-based program to check inconsistency errors in the null-hypothesis significance testing, presented in major psychology journals.

According to the statistics of WebDataCommons, the proportions of entity type, relational type and matrix type of tables in web pages are 59.7%, 38.6% and 1.3%, respectively [9]. However, in vertical domains such as the field of finance, most of tables are matrix-type and have explicit or implicit hierarchical headers. There are some studies about recognizing this type of tables. Fang et al. [3] proposed a Random Forest classification to identify the complex headers in tables; Nagy et al. [11] leveraged rule-based method to extract data categories and data hierarchies from table headers. Based on the extracted tables, there are many understanding tasks, such as linking text to table cells [6], table cell search for a given query [15], ad hoc search over tables [18], transforming complex tables to the form that can be stored in a database [14]. Our task, cross-checking over numerical tables, is also a table understanding task based on extracted table structure.

7 CONCLUSION

When investor finds out accounting errors in financial reports, they may doubt firm's governance capacity and authenticity of firm's accounting information. In this paper, we aims at automatic numerical cross-checking over tables in a document, and propose an end-to-end solution to detect whether two table cells are semantically equivalent or not. Based on this model, an auditing tool is built to support guided cross-checking. This system has been widely adopted in the Chinese financial community. Users feedback and theoretical analysis confirm that it helps to save around 50% time.

8 ACKNOWLEDGEMENTS

This work was supported by the National Key Research and Development Program of China under Grant No. 2017YFB1002104, the National Natural Science Foundation of China under Grant No. U1811461, and the Innovation Program of Institute of Computing Technology, CAS.

REFERENCES

[1] Yixuan Cao, Hongwei Li, Ping Luo, and Jiaquan Yao. 2018. Towards Automatic Numerical Cross-Checking: Extracting Formulas from Text. In *WWW*.
[2] Preeti Choudhary, Kenneth J Merkley, and Katherine Schipper. 2019. Do Immaterial Error Corrections Matter? *Available at SSRN 2830676* (2019).
[3] Jing Fang, Prasenjit Mitra, Zhi Tang, and C Lee Giles. 2012. Table header detection and classification. In *AAAI*.
[4] Vivian W Fang, Allen H Huang, and Wenyu Wang. 2017. Imperfect accounting and reporting bias. *Journal of Accounting Research* (2017).
[5] Naeemul Hassan, Fatma Arslan, Chengkai Li, and Mark Tremayne. 2017. Toward Automated Fact-Checking: Detecting Check-worthy Factual Claims by ClaimBuster. In *KDD*.
[6] Dae Hyun Kim, Enamul Hoque, Juho Kim, and Maneesh Agrawala. 2018. Facilitating document reading by linking text and tables. In *UIST*.
[7] Mio Kobayashi, Ai Ishii, Chikara Hoshino, Hiroshi Miyashita, and Takuya Matsuzaki. 2017. Automated Historical Fact-Checking by Passage Retrieval, Word Statistics, and Virtual Question-Answering. In *IJCNLP*.
[8] Alastair Lawrence. 2013. Individual investors and financial disclosure. *Journal of Accounting and Economics* (2013).
[9] Oliver Lehmberg, Dominique Ritze, Robert Meusel, and Christian Bizer. 2016. A large public corpus of web tables containing time and context metadata. In *WWW*. 75–76.
[10] Moin Nadeem, Wei Fang, Brian Xu, Mitra Mohtarami, and James R. Glass. 2019. FAKTA: An Automatic End-to-End Fact Checking System. In *NAACL*.
[11] George Nagy and Sharad Seth. 2016. Table headers: An entrance to the data mine. In *ICPR*.
[12] Michèle B. Nuijten, Chris H. J. Hartgerink, Marcel A. L. M. van Assen, Sacha Epskamp, and Jelte M. Wicherts. 2016. The prevalence of statistical reporting errors in psychology (1985–2013). *Behavior Research Methods* (2016).
[13] Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*.
[14] Alexey O Shigarov, Viacheslav V Paramonov, Polina V Belykh, and Alexander I Bondarev. 2016. Rule-based canonicalization of arbitrary tables in spreadsheets. In *ICIST*.
[15] Huan Sun, Hao Ma, Xiaodong He, Wen-tau Yih, Yu Su, and Xifeng Yan. 2016. Table cell search for question answering. In *WWW*.
[16] James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a Large-scale Dataset for Fact Extraction and VERification. In *NAACL*.
[17] Andreas Vlachos and Sebastian Riedel. 2014. Fact Checking: Task definition and dataset construction. In *Workshop on Language Technologies and Computational Social Science, ACL*.
[18] Shuo Zhang and Krisztian Balog. 2018. Ad hoc table retrieval using semantic similarity. In *WWW*.