Recognising Tabular Mathematical Expressions Using Graph Rewriting

Mohamed Alkalai

School of Computer Science, University of Birmingham
M.A.Alkalai@cs.bham.ac.uk

Abstract. While a number of techniques have been developed for table recognition in ordinary text documents, very little work has been done on tables that contain mathematical expressions. The latter problem is complicated by the fact that mathematical formulae often have a tabular layout themselves, thus not only blurring the distinction between table and content structure, but often leading to a number of possible, equally valid interpretations. However, a reliable understanding of the layout of a formula is often a necessary prerequisite to further semantic interpretation. In this paper, a graph representation for complex mathematical table structures is presented. A set of rewriting rules is applied to the graph allows for reliable re-composition of cells in order to identify several valid table interpretations. The effectiveness of the technique is demonstrated by applying it to a set of mathematical tables from standard text books that has been manually ground-truthed.

1 Introduction

The matrices of cells could be considered as the simplest tables: There are no spanning cells through columns or through rows. The borders of all the cells are marked by the ruling lines. This kind of tables is easy to recognise using the graphic ruling lines. However, due to the lack of standard convention of composing tables, not all tables follow such a distinction. As for the physical layout, it can be noted often the presence of cells that spread over several lines or several columns, and sometimes the borders of neighbouring cells are even misaligned. Also, in the majority of cases, the borders and the rules of a table are not marked by the graphic lines.

To characterize the table structure for various domains of documents, a flexible framework representation is necessary. The table's syntactic layout and the semantic structure must be depicted. While the information about the physical layout can contribute to table re-composition, the logical structure can be used to extract the table's content for re-use purposes.

Ramel et al. [6] analyse the most two well-known table representation systems (which are introduced by World Wide Web Consortium (W3C) and Advancement of Structured Information Standards (OASIS)) that are used to represent tables and find that they share the same deficiencies. First, the representation of irregular physical layouts are difficult. The poorly aligned borders of cells are

J. Ruiz-Shulcloper and G. Sanniti di Baja (Eds.): CIARP 2013, Part II, LNCS 8259, pp. 124–133, 2013.

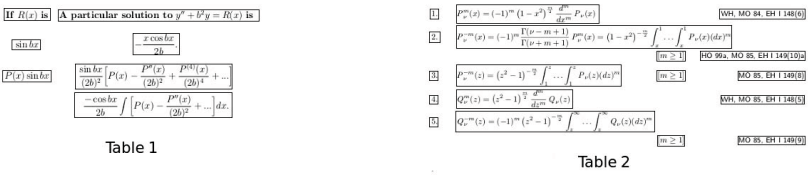

Fig. 1. Cell identification with tables containing multiline expressions that are taken from [5]

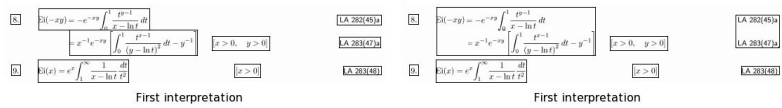

Fig. 2. Two different interpretations of a single table that is taken from [5]

not allowed and improvised solutions are provided for the spanning cells. Finally, limited means are supplied for the description of the logical structure of a table.

The aim of this work is to develop a table recognition algorithm that is particularly good for tables containing mathematical expressions. As the distinction between tables and complex typeset mathematical formulae spanning multiple lines is often difficult, the narrow definition of tables is forgone and instead consider a far wider range of expressions as tables as is usually the case in the literature.

The tabular form in which many mathematical expressions are being presented can often lead to ambiguities in the interpretation to what essentially constitutes a table component (i.e., a column, row or cell). While in table understanding of ordinary text tables [7], [8], the goal is generally to restrict a result to a single valid interpretation, for mathematical tables these ambiguities can lead to several possible valid interpretations. Therefore, the aim of the proposed recognition procedure is to produce as a result the set of possible valid interpretations.

Since, as mentioned above, that there is no standard convention of composing tables and that there is a need of building table representation framework which is flexible enough to deals with tables from various domains. Therefore, The framework that is illustrated in section 3 is constructed based on abstract concepts that allow for producing the maximum cells, columns and rows which can be extracted from table form. Graph rewriting rules are also introduced in this framework to selectively utilized for recomposing table's cells. The nature of the proposed framework gives the opportunity to used it on different table structures from various area of sciences like Literacy, Mathematics, Physic, Chemist...etc.

2 Interpretation of Mathematical Tables

While some tables, for example, Cayley tables in abstract algebra, are quite straight forward to recognise due to their easy tabular composition and some-

times clear separation of rows and columns with bars, this is generally not the case. In fact, the common absence of any vertical or horizontal bars as well as the complexity of formulae often spanning multiple lines make it not only difficult to identify the cell structure but can lead to a number of different interpretations for the same table, which are often equally valid.

Figure 1 presents two tables taken from [5] with a fairly conservative column and row layout. There is indeed a unique ideal interpretation for Table 1, consisting of two columns and three rows, where the cell in the lower right hand corner contains a math expression spanning two lines. In addition, given the difference in font weights one could even interpret the first line as a clear header row.

Table 2 on Figure 1 is less straightforward given the overlapping expressions in the second line. However, one can still come up with a unique interpretation of five rows and three columns. However, due to the overlap of the formulae which are effectively in a column of their own, it is difficult to obtain this interpretation automatically.

Figure 2 presents a clipping from a more complex table also taken from [5]. Here it is possible to see two different interpretations, both with their own merits. While both interpretations regard the basic table as consisting of four columns, the first interpretation results in three rows, using the formula names on the right hand side as header column. The second interpretation on the other hand uses the enumeration in the first column as header. Obviously there are still more interpretations: For example, one with three columns with the middle column containing complex formulae or even one with only two columns, where the right column contains named multiline formulae that possibly even be considered as subtables.

This gives not only rise to the problem of finding a method that can yield a number of possible valid interpretations, but also the need to finding an adequate grid structure to represent such tables holding the different interpretations and to give a means to re-compose the recognised cells.

3 Multi-interpretations of Table's Re-composition

In this section a description of the proposed method is given. The input of the technique is the bounding box of table cells which are extracted by the method presented in [1]. Using these cells, the algorithm first produces the maximum columns and also the maximum cells in each column that can be extracted from a table. This is further described in the preprocessing steps section below. Then an initial graph that represents the table grid and the relationship between their nodes (cells) is defined and built. Also a new set of graph rewriting rules that are used to produce all possible valid interpretations is illustrated. Finally, experimental results on 150 tables are shown and evaluated.

3.1 Preprocessing Steps

Several definitions are given below to formally describe the concepts of how the maximum columns and cells from tables are extracted. The first definition is for the Bounding Box of the cell component C:

Definition 1 (Bounding Box). *Let c be a cell, then the borders of its* bounding box *are defined by* $l(c), r(c), t(c), b(c)$ *representing left, right, top and bottom limit respectively where* $l < r$ *and* $t < b$

Before building a graph to represent table structure, all cells C are first sorted ascendantly using $l(c)$. Then, initial columns are constructed by splitting C on the cell that does not horizontally overlap with all cells which are above it.

Definition 2 (Initial Columns). *Let* $C = \{c_1, c_2, ...c_n\}$ *be all cells ordered such that* $l(c_1) < l(c_2)$ *and col be a column of table. Then* $col = \{c_1, c_2, ...c_m\}$ *if one of the set* $[r(c_1), r(c_2), r(c_3), ...r(c_m)] < l(c_{m+1})$ *where* $n = 1, 2, ...$ *and* $m < n$

In case there is an absence of cell which should be beneath or above a cell that is being checked using the step described above, a virtual cell c' is added. When the graph is built later, these virtual cells are represented as nodes. The goals of adding such nodes to the graph are firstly to avoid the complex relationship between nodes and secondly to use such nodes to detect actual rows. Some examples are shown in Figure 3

In order to locate the position of virtual cells, the definition 3 in [1] is recalled to calculate the borders of lines and then use them to detect if there is an existence of a line which has no corresponding cell (belong to a particular column) vertically overlapped with its borders. If so, a virtual cell is added.

Definition 3 (Virtual cells). *Let* $col = \{c_1, c_2, ...c_n\}$ *be a column and* $l = \{g_1, g_2, ...g_n\}$ *be a line such that if* $b(c_1) =< t'(l_1) || t(c_1) >= b'(l_1)$ *where* $t'(l_1) = \min_{g \in l_1} t(g)$ *and* $b'(l_1) = \max_{g \in l_1} b(g)$ *then add a virtual cell* c'_1.

3.2 Tabular Representation Using Graph Model

The total cells which are produced from the above steps are utilized to build an initial graph that represents the table structure. Each node N in this graph

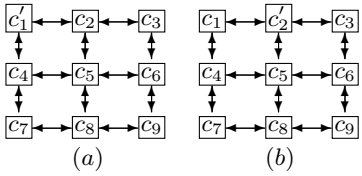

Fig. 3. Examples of virtual cells which their borders appear to be bigger

G which corresponds to a cell c has four edges E with four directions where l, r, t and b are labelled left, right, top and bottom edge direction respectively (an exception is for border nodes which might have only two or three edges). Also, there must be an existence of all possible first degree connections between nodes N. The first degree connections mean here the edges that directly connect a node n with its adjacent nodes.

Definition 4 (Graph Specifications). *Let n be a node which represents a cell c, then the directions of its outgoing edges are defined by $l(n), r(n), t(n), b(n)$ representing left, right, top and bottom directions respectively. Let n' be directly adjacent node to n at any direction such that every $l(n)$ there exists of $r(n')$ and likewise for $r(n), t(n), b(n)$.*

1) Type of Nodes: When constructing the initial graph, one can divide the nodes to four types. The classification process is done by checking whether there is a horizontal overlap between columns or not. Table 1 shows the node types and how they are treated by the interpretor.

Table 1. Type of nodes

Node Types	Definition
R	is a real node which must not be merged with other nodes from other columns
V	is a virtual node which must not be merged with other nodes from other columns
R^*	is a real node which can be merged with other nodes to form one of the possible valid table interpretations
V^*	is a virtual node which can be merged with other nodes to form one of the possible valid table interpretations

2) Type of Relationships between nodes (Edges): To avoid having complex relationships between nodes, a graph which represents the maximum number of nodes for a table is constructed. This provides us with simple relationship between nodes which are horizontal and vertical edges.

3.3 Construct Initial Graph (Example)

The graph in Figure 4 represents Table 2 in Figure 1 which illustrates one possibly valid interpretation of the table. As can be seen, the proposed algorithm succeeded in distinguishing the second column that represents equations from the misaligned third column that represents the conditions associated with these equations and eventually splits them to two columns.

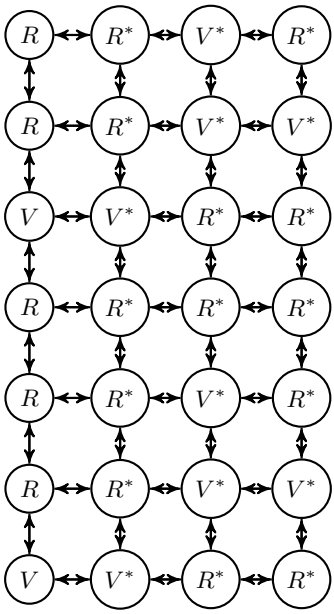

Fig. 4. A graph represents one of the possible interpretations of the table's columns shown in table 2 on the right of Fig 1

3.4 Graph Rewriting Rules

Graph rewriting rules are composed to represent structural information of table form. The graph defined above is used to represent them. Although, in [3] and [2] the authors have used graph rewriting rules before to analyse table layout, due to the complex structure of the table domain that are used in the experiments, new rules are produced.

Let N and E represent a specific set of nodes and a specific set of relationships between nodes called edges respectively. Then, A graph rewriting rule can

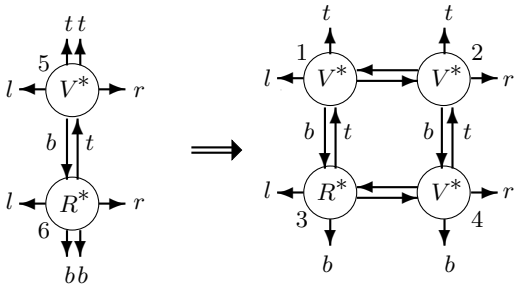

Fig. 5. Example of production rule

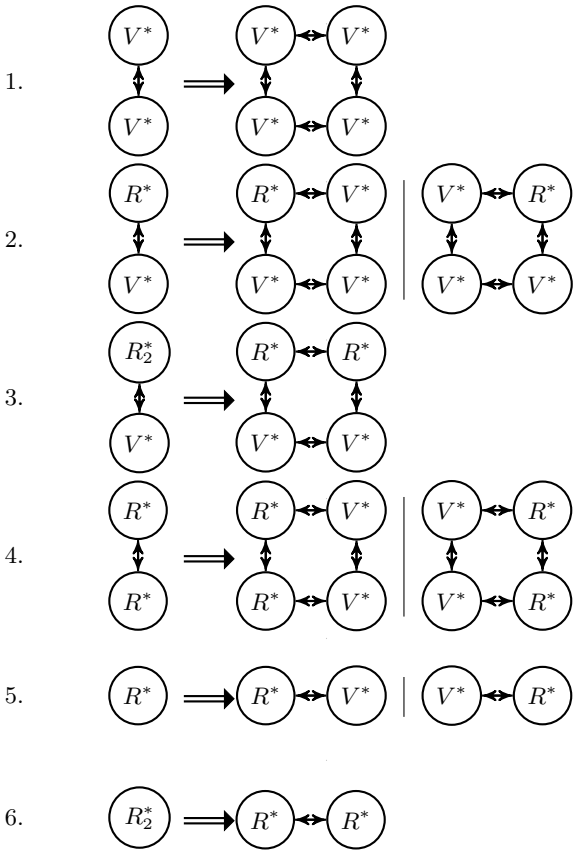

Fig. 6. Full production rules

be represented by the following tuple $g = \{N, E, P\}$ where P are rewriting production rules which has the form $lhs \rightarrow rhs$, this specifies two graphs where the subgraph rhs in a host graph (G) can be replaced with a graph lhs. The embedding relations ER associated with each rewrite rule $lhs \rightarrow rhs$ specify how the new subgraph lhs is connected to the remainder graph of the host graph G, after rhs is removed. The notation containing four-tuples of the form $\{(n_1,e_1,n_2,e_2)$: $n_1,n_2 \in N$; $e_1,e_2 \in E\}$ is used to represent embedding relations ER. Figure 5 shows an example of production rule.

Embedding rule ER which tells edge label conversion from rhs to lhs for the production rule showed in Figure 5 is expressed as follows:

$ER = ((1V^*, l, 5V^*, l), (1V^*, t, 5V^*, t), (2V^*, r, 5V^*, r),$
$(2V^*, t, 5V^*, t), (1V^*, b, 5V^*, b), (3R^*, t, 6R^*, t),$
$(3R^*, l, 6R^*, l), (4V^*, r, 6R^*, r), (3R^*, b, 6R^*, b),$
$(4V^*, b, 6R^*, b))$

3.5 Full Table Production Rules

A sample of the production rules that are needed to represent all possible table interpretations are shown in Figure 6. Due to the pages number limitation, illustration of all rules which fully cover the different cases of node combinations is not possible. By using these rules, one can produce all possible table interpretations.

4 Evaluation and Experimental Results

To accomplish the table recognition evaluation, preparing table ground-truthing is usually needed. In [4] the author stated that, in some cases, the researchers who are ground-truthing tables might have different opinions about the right way of ground-truthing a table. Sometimes, several interpretations seem to be justifiable and appear to be equally valid. Taking into account this fact and for evaluation purposes, 150 tables were manually ground-truthed, such that, each table has all possible interpretations that can be extracted from it. To facilitate the comparison procedure, a visual technique is designed which allows us to visually assess the table recognition output. The technique draws rectangles around table cells. Each column's cells are given a unique colour to their rectangles. Experiments are done using 100 pages taken from [5] which contains 150 tables. Table 2 illustrates concise information about the experimental results. In this table, the 150 tables are categorised to three groups based on the number of possible interpretations that can be obtained from a table. This can be accomplished using the ground-truth tables. A comparison between outputted table possible interpretations and the corresponding ground-truth table is then manually done. The results of this comparison are classified into three categories. This is determined by observing how far one possible interpretation of the table from the proposed system matches one possible interpretation of the table according to the ground truth set. These three categories are: 1) Table interpretations that completely and correctly extracted. An output table is classified under this category if it 100% matches. 2) Table interpretations partially extracted. Here the

Table 2. Results of applying the proposed table interpretation technique on 150 tables

No. of Tables	Ground Truth Table Dataset	Output Table Dataset		
	Number of Possible Table Interpretations	No. of Tables Interpretations Completely and Correctly Extracted	No. of Tables Interpretations Partially Extracted	No. of Tables Interpretations That are missed
82	2	124	26	14
65	4	141	56	63
3	7	6	7	8

matching rate is approximately between 75% and 95%. 3) Tables interpretations that are missed. In this cases, the matching rate is 0%.

4.1 Analysis of Table Interpretations That Are Partially Extracted or Missed

Although the experimental results — presented in Table 2 — show already a promisingly high accuracy rate, there is still a considerable problem with the mis-clustering of some cells to the wrong column (in the case of table interpretations that are partially extracted) and the failure of splitting two columns (in the case of table interpretations that are missed). An analysis of these cases yields that the majority of mis-clustering and failure cases are due to the preprocessing step when it fails to assemble the cells into their proper columns. One possible approach to tackle this kind of problem and eventually improve the accuracy of the proposed approach is to manually intervene by adding marks on tables which assist the proposed method in inferring the correct column and as a result, extract the all possible valid table interpretations. Figure 7 illustrates how adding marks on tables improves the accuracy rate.

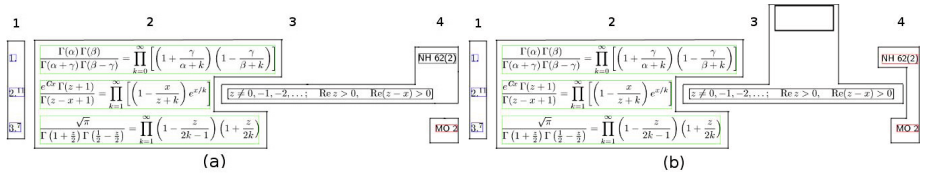

Fig. 7. Example of table re-composition (a) before manual intervention (b) after manual intervention

The Figure 7 shows tables before and after the manual intervention. Each column in this table is bordered and given a number. By observing table (a) it can be clearly seen that last cell in the first row is wrongly clustered to the third column where it should have been gathered with the cells in fourth column. Table (b) shows a solution of the problem by adding an empty rectangle over the third column to tell the proposed method that the last cell in the first row is not overlapped with all cells in third column and therefore, it should be gathered with cells in fourth column.

5 Conclusion

The proposed framework introduced in this paper was built accordingly upon the observation of a wide range of tabular forms which occur in many documents from different domains. The abstract components of this framework can be used as basis of wide range of other applications of document recognition.

The technique represented here is able to produce several interpretations of table form. Unlike other table representation techniques, the proposed approach has the capability to deal with misaligned columns that sometimes appear in tabular mathematical components. Adding virtual nodes to the initial graph prevents complex relationship between nodes and would contribute in deciding the actual table's rows. Using the described production rules allow for producing more than one possible valid interpretations of table structure. The experiments in 150 tables show promising results.

References

1. Alkalai, M., Sorge, V.: Issues in mathematical table recognition. In: Conferences on Intelligent Computer Mathematics (CICM 2012), MIR Workshop (2012)
2. Amano, A., Asada, N.: Graph grammar based analysis system of complex table form document. In: ICDAR, pp. 916–920. IEEE Computer Society (2003)
3. Cooperman, R., Armon Rahgozar, M.: A graph-based table recognition system. In: SPIE Proc., pp. 192–203 (1996)
4. Hu, J., Kashi, R.S., Lopresti, D.P., Wilfong, G.T., Nagy, G.: Why table ground-truthing is hard. In: ICDAR, pp. 129–133 (2001)
5. Jeffrey, A., Zwillinger, D.: Table of Integrals, Series, and Products. Elsevier Inc. (2007)
6. Ramel, J., Crucianu, M., Vincent, N., Faure, C.: Detection, extraction and representation of tables. In: Proceedings of the Seventh International Conference on Document Analysis and Recognition, vol. 1, pp. 374–378. IEEE Computer Society, Washington, DC (2003)
7. Costa Silva, A., Jorge, A.M., Torgo, L.: Design of an end-to-end method to extract information from tables. International Journal Document Analysis Research 8(2), 144–171 (2006)
8. Zanibbi, R., Blostein, D., Cordy, J.R.: A survey of table recognition: Models, observations, transformations, and inferences. Int. J. Doc. Anal. Recognit. 7(1), 1–16 (2004)