Advances in Computer, Signals and Systems (2024)
Clausius Scientific Press, Canada

DOI: 10.23977/acss.2024.080307
ISSN 2371-8838 Vol. 8 Num. 3

Research on an Automatic Table Generation Model Based on Entity Attribute Relations

Maolong Teng[1,a], Zhinan Lin[2,b,*], Chaoyue Liu[3,c]

[1]Hunan University of Science and Technology, Xiangtan, China
[2]Hunan Vocational Institute of Technology, Xiangtan, China
[3]Guizhou Provincial Transportation Planning Survey and Design Research Institute Co., Ltd., Guiyang, China
[a]1929931622@qq.com, [b]409435402@qq.com, [c]464683877@qq.com
[*]Corresponding author

Keywords: Universal table model, Entity attribute connections, Automatic table generation, Tree structure

Abstract: Aiming at the problem that existing automatic table generation technologies cannot dynamically adjust the structure and content of tables based on real-time changes in data or user needs, a universal table model for automatic table generation is proposed. This model takes entity attribute relationships as the design core and uses a tree structure to describe the structure of tables and the relationships between cells. This tree structure is defined as a multi-dimensional relationship oriented table structure tree, emphasizing the guiding role of entity attribute relationships in the table generation process, aiming to improve the efficiency and accuracy of table automatic generation. After application testing and experimental comparison in actual projects, it has been proven that the model has good universality and practical value.

1. Introduction

A table is a compact, well formatted, and easy to visually display form of data. It is not only a visual communication mode, but also a tool for organizing and organizing data. Tables are widely used in the daily affairs of enterprises and governments, usually used to organize and display various types of data. By using tables, readers can easily understand and analyze information.

Automatic table generation refers to the process of using computer programs or tools to automatically create or generate tables based on input data or conditions. Through algorithms and automation technology, automatic table generation can save time, reduce manual labor, improve efficiency, and ensure accuracy and consistency. In terms of table generation, Zhang Chang[1]proposed the theory of splitting and reorganizing multi-source analogies to automatically generate tables. This method divides the table into matrices, and the sub matrices in the table are derived from the sub matrices in other tables and merged together. Zhang Jing[2]designed an automatic report generation program, which needs to define the content of the table header before making the table. It is mainly used for research on soil and water resource management. Zhang[3]proposed the task of dynamic table generation, which uses term based entity sorting with

multiple retrieval models and iterative representation generation algorithms. Given a query, a relationship table containing relevant entities and their key attributes is generated.

At present, rule-based and template based methods require manual annotation, and corresponding templates must be created for each generated table. After the table structure is determined, it cannot be flexibly modified, and its application scope is limited. Additionally, data stability is poor. When the data source changes, the data in the table cannot be updated in real time, and additional mechanisms need to be established. Based on machine learning and deep learning methods, text generation tables can be automatically generated, but only simple three line tables can be generated, and the actual content may not meet the requirements and expectations, making it impossible to directly generate tables from the database data in an ideal way according to the requirements. The scope of application is narrow and the algorithm accuracy is not high.

In response to the current problems in automatic table generation, this article proposes a table automatic generation model based on entity attribute connections, based on the analysis of table structure features and the correlation between cells. This model uses a multidimensional relationship oriented table structure tree to represent the structure of tables and the relationships between data. Using this model can effectively handle situations where there is a large amount of table data or frequent updates of table content are required, and ensure the accuracy of the data.

2. Entity Attribute Connection

A table contains a large amount of relational data, whose content is usually associated with entities in the database. Entities refer to objectively distinguishable things. In the database, entities are tables, and each row of data in the table is an instance of the entity. A set of entities of the same type is called an entity set[4]. Attributes refer to data items that describe the properties or features of an entity. An entity can have multiple attributes, and the specific values taken by attributes are called attribute values. The range of all values that an attribute may take is called the domain of that attribute. For example, the domain of the "land name" attribute for fill soil is plain fill, miscellaneous fill, fill soil, and waste residue; The domain of fault attribute "fracture zone components" is fault gouge and breccia. The types of attributes can be integer, entity, string, etc.

According to the Pattern classification, entities are mainly divided into the following types:

(1) Main entity: an independent entity that does not rely on others.

(2) Sub entity: Derived from the main entity, with more detailed or specific attributes than the main entity.

(3) Attribute type entity: describes a certain aspect or feature of the main entity.

(4) Associated entity: usually represents other entities that have an associated relationship with the main entity.

Relationships refer to the set of associations within or between entities, the connections within entities refer to the connections between various attributes within entities, and the connections between entities refer to the connections between different sets of entities. There are three common types of relationships between entities: one to one, one to many, and many to many.

(1) One to one: Each entity in entity set A, at most one entity in entity set B is associated with it, and vice versa.

(2) One to many: For each entity in entity set A, there are n entities ($n \geq 0$) associated with entity set B. Conversely, for each entity in entity B, there is at most one entity associated with entity set A.

(3) Many to many: For each entity in entity set A, there are n entities ($n \geq 0$) associated with entity set B. Conversely, for each entity in entity B, there are also m entities ($m \geq 0$) associated with entity set A.

3. Entity Attribute Table Model

In order to meet the diverse needs of different scenarios for table styles, it is necessary to set the table structure based on specific strategies and accurately fill the structured information data stored in the database into the table according to its inherent correlation relationships. A universal table model for automatic table generation has been proposed, which takes entity attribute relationships as the core of design. Based on the tree structure, the table structure and the organization of table data have been redesigned. This tree structure is called a multidimensional relationship oriented table structure tree. The multidimensional relationship oriented table structure tree not only comprehensively describes various complex types of table structures, but also has good scalability and can adapt to constantly changing table requirements. Through this model, corresponding table formats and layouts can be generated according to different display requirements, and the relationships between data can be accurately represented. The generated tables can be flexibly adjusted and optimized according to specific data content and display requirements.

3.1. Formal Expression of Tables

3.1.1. Table Structure

A table is a tool that presents related information in a concise manner. With its unique structural layout, it can cleverly express the two-dimensional or even multidimensional logical relationships between data, becoming a powerful assistant for expressing data [5]. The structure of a table can be specifically divided into two parts: layout structure and logical structure. Layout structure refers to the physical information such as the position, size, width, etc. of the cells that make up the table. A well structured table can clearly and intuitively display data, helping readers quickly understand the content of the table. And logical structure refers to the organization and correlation of data within a table, which is a deeper level of content. Tables are a common usage for representing relational data, where data is typically represented in the form of attributes and their values. In order to present data more effectively in a two-dimensional space, tables typically use a hierarchical structure. In order to better analyze the structure of tables and the hierarchical dependency relationships in table data, relevant concepts in tables have been defined as follows:

Definition 1 (Attribute List Cell) The cells located at the top of the table are the Chinese field names for the data described in the table. These cells are used to identify the names of the entities or attributes represented by each column in the table.

Definition 2 (Attribute Value Cells) The cells corresponding to the attribute list cells in the table that require specific data to be filled in.

Definition 3 (Indicating domain) A continuous range in which all cells are attribute list cells, located in the first n rows of the table, where $1 \leq n < p$, p is the total number of rows, used to describe the category and properties of the data represented by the table.

Definition 4 (Data Domain) A continuous area located below the indicator field, where all cells are attribute value cells, storing the true data of the categories and attributes described in the indicator field.

Definition 5 (Table) Table $TB = \{(h_{1,1} \ldots, h_{1,n}), \ldots, (h_{k,1}, \ldots, h_{k,n}), (d_{k+1,1}, \ldots, d_{k+1,n}), \ldots, (d_{k+m,1}, \ldots, d_{k+m,n})\}$, where each $h_{i,j}(1 \leq i \leq k, 1 \leq j \leq n)$ represents the data content in the attribute list cell within the table indicator field, and each $d_{i,j}(k+1 \leq i \leq k\_m, 1 \leq j \leq n)$ represents the data content in the attribute value cell within the table data field.

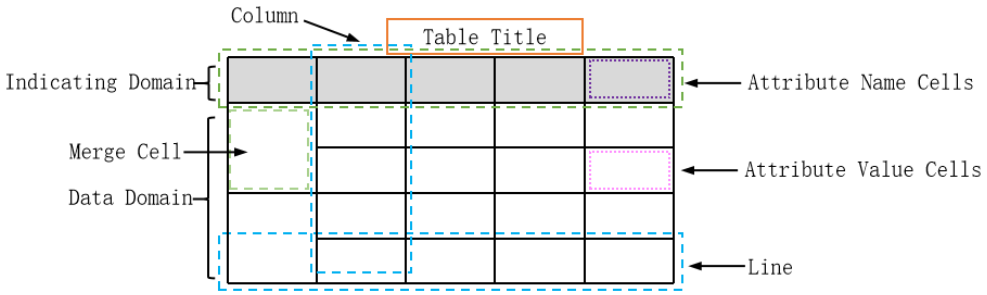

Figure 1: Structure diagram of the table.

As shown in Figure 1, it is a simple table structure diagram. Firstly, based on the indicator function and display function of the table, it is divided into attribute list cells and attribute value cells. Then, the area composed of attribute list cells is named as the indicator field, and the area composed of attribute value cells is named as the data field. This divides the table into two areas. Through the table structure, the hierarchical dependency relationship between the data in the table can be expressed, which has a good indicator effect for analyzing the hierarchical relationship between the data content in the table later.

3.1.2. The Correlation Between Table Cells

The association between table cells refers to the direct or indirect association or connection formed between each cell in a table based on the data content they store and their relative positions in the table structure. This kind of correlation can reflect the similarity, correlation, hierarchical structure, or other logical relationships between data, which helps to have a more comprehensive understanding and analysis of the data in the table. The cells in the indicator field and data field of a table are not isolated and have semantic significance in conveying the logical relationships between cells. There is a certain semantic correlation. From a logical structure perspective, a table is a data storage structure composed of a set of cells C and a set of relationships R between cells. The cell set C is:

$$C = \{c | c \in T_1 \cup T_2\} \tag{1}$$

In the formula, $T_1 = \{Header\}$, $T_2 = \{Data\}$. Data represents the attribute value cell, and Header represents the attribute list cell. The relationships between cells can be divided into three types: hierarchical relationships (represented as Hier), indicative relationships (represented as Index), and master-slave relationships (represented as Master).

Definition 6 (Hierarchy) The relationship between each attribute list cell in the table can be divided into multiple levels, each level consisting of an indefinite number of attribute list cells, and the sum of the widths of the attribute list cells at each level is equal. The content in the meta cells of the attribute list between levels has a hierarchical membership relationship from bottom to top.

Definition 7 (Indicative Relationship) The attribute list cells in the table provide semantics and explanations for attribute value cells, which can reflect the data type and format of attribute value cells. The attribute list cell has a one to many relationship with the attribute value cell.

Definition 8 (master-slave relationship) The dependency relationship between the data in each column of the table can be relatively divided into the main column and the subordinate column. The data in the subordinate column is usually associated with the data in the main column, and its content is often influenced by the content of the main column, representing more detailed and specific information under the main column.

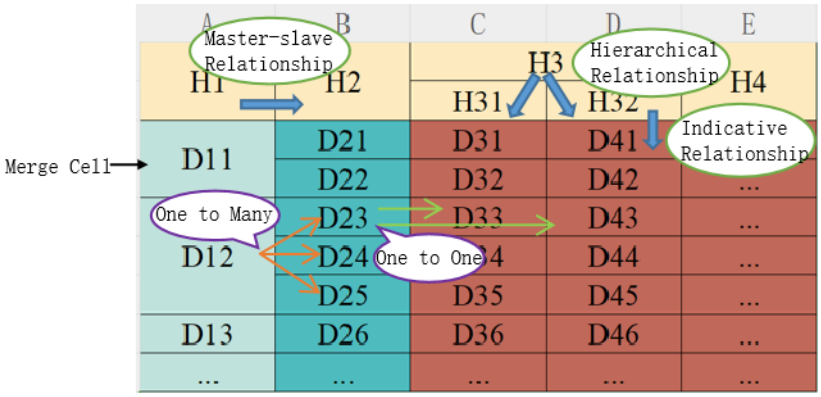

Figure 2: Table cell diagram.

The hierarchical relationship set H is:

$$H = \{(c_\alpha, c_\beta) | (c_\alpha, c_\beta) \in T_1, c_\alpha \xrightarrow{Hier} c_\beta\} \tag{2}$$

The hierarchical relationship is the relationship between the Header, which indicates the level of the meta cell in the attribute list. $c_\alpha \xrightarrow{Hier} c_\beta$ indicates that the level of element c_α in the attribute list is higher than that of element c_β in the attribute list. For example, the hierarchy of cell H3 in Figure 2 is higher than that of cells H31 and H32. The indicator relationship set I is:

$$I = \{(c_\alpha, c_\beta) | c_\alpha \in T_1, c_\beta \in T_2, c_\alpha \xrightarrow{Index} c_\beta\} \tag{3}$$

The indication relationship is the relationship between the Header and the Data. $c_\alpha \xrightarrow{Index} c_\beta$ indicates that the attribute list cell c_α indicates the attribute value cell c_β. For example, cell H32 in Figure 2 indicates the data content type of cells D41, D42, and so on. The set of master-slave relationships A is:

$$A = \{(c_\alpha, c_\beta) | c_\alpha \in T_1, c_\beta \in T_2, c_\alpha \xrightarrow{Master} c_\beta\} \tag{4}$$

This article breaks down tables into columns for analysis, which can group information data and better handle the relationships between columns. The master-slave relationship is the relationship between Data columns. $c_\alpha \xrightarrow{\text{Master}} c_\beta$ indicates that the data in column c_α depends on the data in column c_β. For example, the data in column B in Figure 2 depends on the data in column A. Specifically, it can be seen that the data in cells D23, D24, and D25 in column B depends on the data in D12, belonging to a one to many relationship.

3.2. Table Automatic Generation Model

3.2.1. Model Framework

The automatic generation of tables requires solving two major problems: one is the structure of the table, and the other is the data of the table. The business logic processing of generating tables starts from the setting of the table structure. Tables are divided into indicator fields and data fields. The content of the attribute value cells in the data field is the detailed and specific data that the table needs to display. The indicator fields can fully describe the main information that the table needs to display, and the content related to the main content that the table needs to display. By indicating the

attribute list cells in the indicator field, the data types under the corresponding columns and the main meanings expressed can be determined. Therefore, in the process of automatically generating tables, users first need to create a table template through the model to determine the default basic information of the table, such as row height and column width. Then, the user also needs to set the indicator fields of the table and determine the source of the table data. Finally, the model will automatically query the data based on relevant settings and fill the data content into the table template, thereby generating an instance table. The model framework is shown in Figure 3.

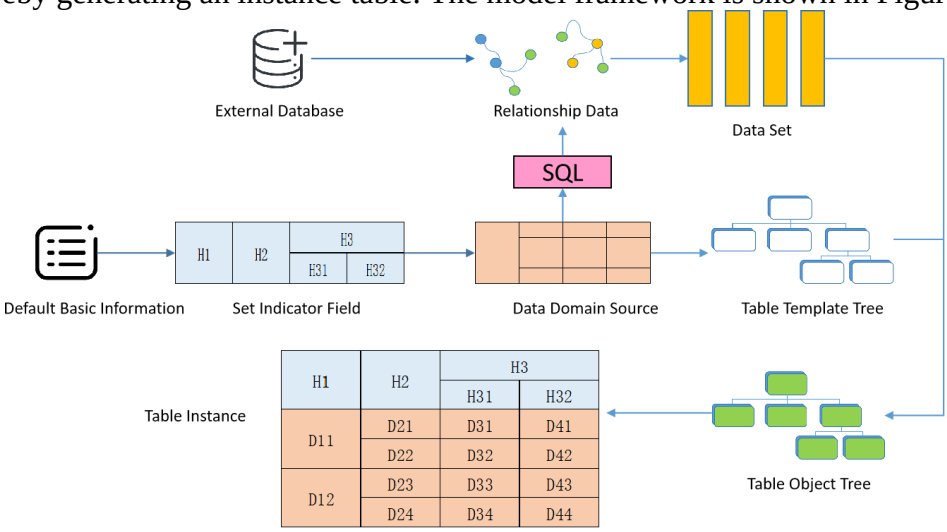

Figure 3: Model frame drawing.

3.2.2. Multidimensional Relation Oriented Table Structure Tree

In the process of generating tables through models, a multi-dimensional relationship oriented table structure tree is proposed for various complex types of tables, emphasizing the guiding role of entity attribute relationships in the table generation process. The multi-dimensional relationship oriented table structure tree can effectively describe the table structure and handle the multi-dimensional relationships between data. The core construction of a multi-dimensional relationship oriented table structure tree is the relationship between objects, which refers to entities and related attributes in the database. Its tree structure is shown in Figure 4.

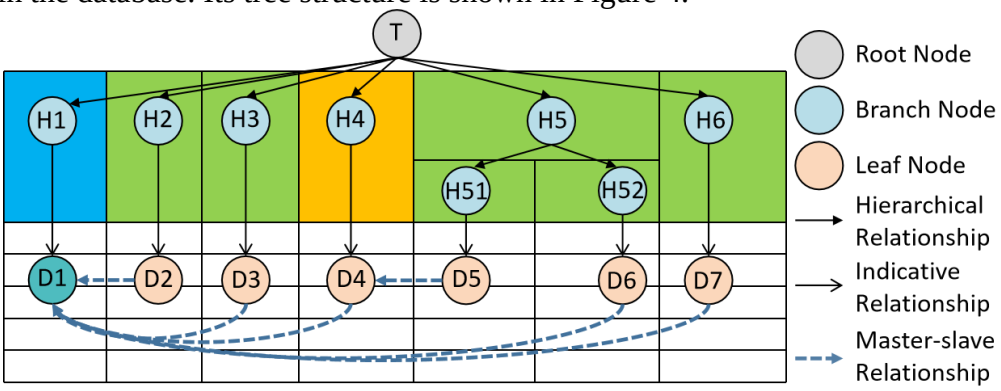

Figure 4: Multidimensional relationship-oriented table structure tree.

From the figure, it can be seen that the nodes in the multi-dimensional relationship oriented table structure tree are divided into three types: root node, branch node, and leaf node. The root node stores the basic information of the table, the branch node is used to store the relevant content of the attribute list cell in its corresponding table indicator field, and the leaf node stores the relevant

content of the attribute value cell in its corresponding column in the data field. The lines in the multi-dimensional relationship oriented table structure tree indicate the association relationship between cells. The multi-dimensional relationship oriented table structure tree can well represent the structure and data content relationship of the table. When generating the table, the attributes and methods set in the node can generate a table that meets practical needs based on the data content.

3.3. Automatic Table Generation Process

Before using the model to generate a table, first check if there is a required table template. If it exists, you can open it directly. If not, you need to create a new table template. When creating a template, users need to first fill in the relevant information of the table, set the content of the indicated fields in the table, and then bind the data fields of each column in the data field based on the entity attribute relationship between tables and the relationship between cells. If the column is a dependent column, it is also necessary to bind the query field associated with the main column. After setting up, click the "Generate" button, and the program will create a model tree. The model generates tables using relevant methods based on the set information. The specific process is shown in Figure 5.

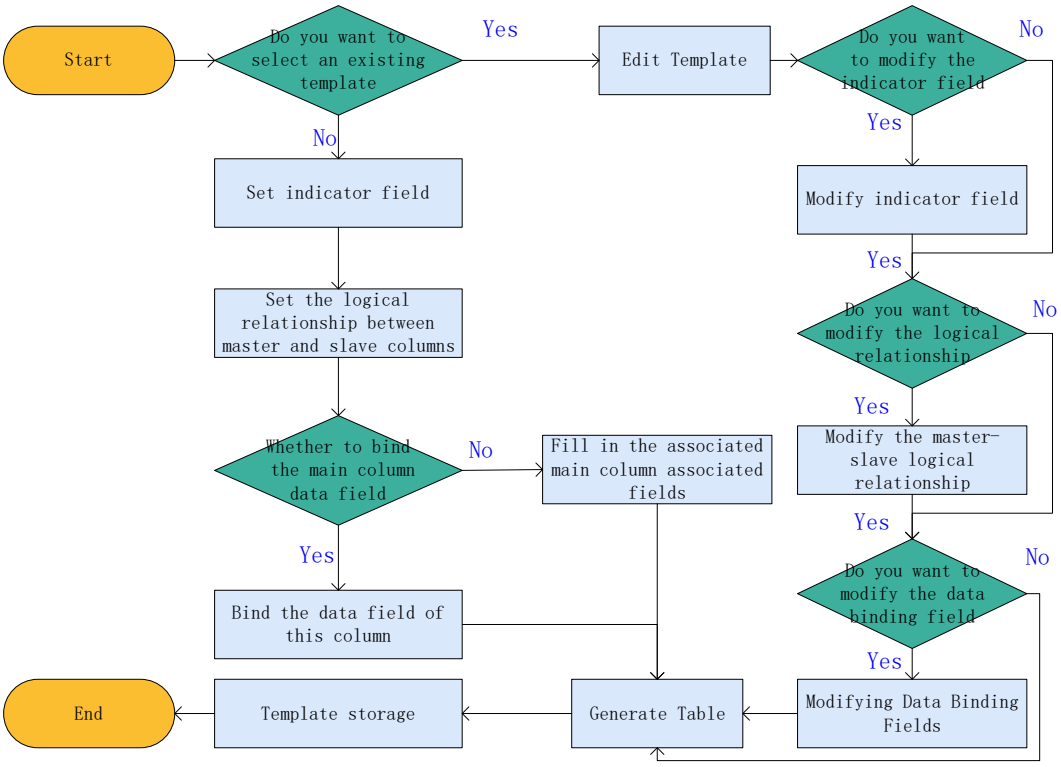

Figure 5: Automatic table generation process.

4. Analysis of Model Application Implementation

This section demonstrates the ability of the proposed table automatic generation model through the use of a highway engineering geological survey system. The highway engineering geological survey system refers to a system that uses networks and information technology to assist in highway engineering geological survey work. The system provides online data collection and management functions, allowing geological survey personnel to directly input and store survey data on site through the network. This can reduce the cumbersome process of traditional paper records and

improve the accuracy and completeness of data. There are over 200 entities and 900 attributes within the system, mainly related to survey tasks, survey points, survey data, and other related information. The project results display requires selecting some content from these information data to create tables for display according to needs. The manual and automatic generation times of 5 types of tables were compared and analyzed, as shown in Figure 6. It can be found that using this model to automatically generate tables significantly shortens the preparation cycle, and the generation time of complex business tables is reduced from 89 minutes to 19 minutes.

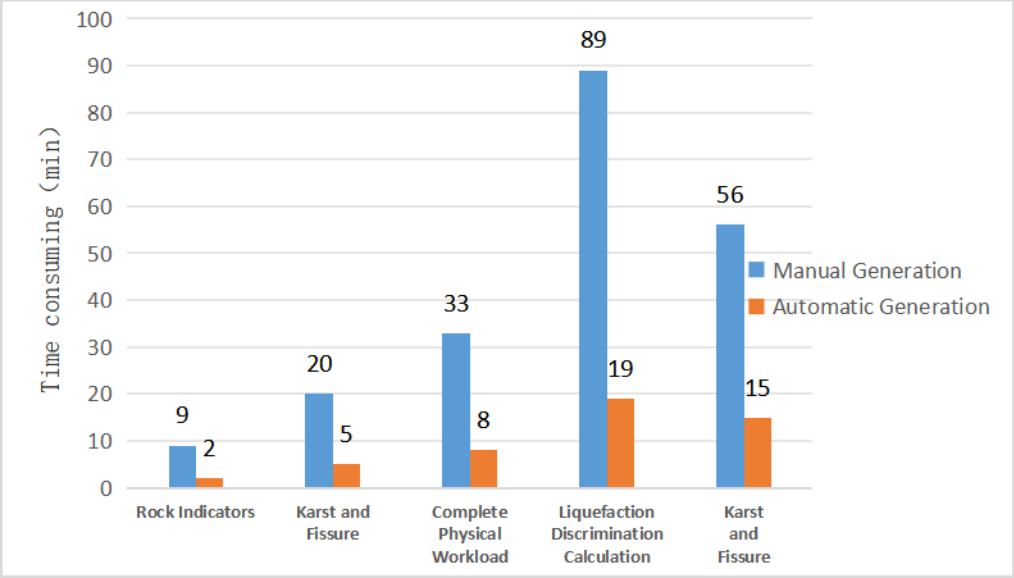

Figure 6: Comparison chart of table generation time.

5. Conclusions

The table automatic generation model based on entity attribute connections proposed in this article can represent the logical relationships between table data by constructing a multi-dimensional relation oriented table structure tree, solving the problems faced by current table automatic generation technology. This model not only improves the accuracy and efficiency of table generation, but also enhances dynamic adjustment capabilities, optimizing large-scale data processing and display capabilities.

References

[1] Zhang Chang, Xu Dongrong, Pan Yunhe. MIS table generation based on multi-source analogy [J]. Computer Research and Development, 1998, (01): 64-69

[2] Zhang Jing, Zhang Yufang. General program design for automatic report generation in soil and water resource management and research [J]. Soil and Water Conservation Bulletin, 1994, (03): 59-63

[3] Zhang S, Balog K. On-the-fly table generation[C]//The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. 2018: 595-604.

[4] Christophides V, Efthymiou V, Palpanas T, et al. An overview of end-to-end entity resolution for big data[J]. ACM Computing Surveys (CSUR), 2020, 53(6): 1-42.

[5] Kim Y S, Lee K H. Extracting table information from the Web[C]//Document Analysis Systems VI: 6th International Workshop, DAS 2004, Florence, Italy, September 8-10, 2004. Proceedings 6. Springer Berlin Heidelberg, 2004: 438-441.