HTML Extraction Algorithm Based on Property and Data Cell

Detty Purnamasari\textsuperscript{1} I Wayan Simri Wicaksana\textsuperscript{2}  
Suryadi Harmanto\textsuperscript{1} Lintang Yuniar Banowosari\textsuperscript{4}  
\textsuperscript{1,2,3}Information Technology \textsuperscript{3}Information Management, Gunadarma University  
Jl. Margonda Raya No. 100 Pondok Cina Depok-Indonesia  
E-mail: \textsuperscript{1}detty@staff.gunadarma.ac.id \textsuperscript{2}iwayan@staff.gunadarma.ac.id  
\textsuperscript{3}suryadi@staff.gunadarma.ac.id \textsuperscript{4}lintang@staff.gunadarma.ac.id

Abstract. The data available on the Internet is in various models and formats. One form of data representation is a table. Tables extraction is used in process more than one table on the Internet from different sources. Currently the effort is done by using copy-paste that is not automatic process. This article presents an approach to prepare the area, so tables in HTML format can be extracted and converted into a database that make easier to combine the data from many resources. This article was tested on the algorithm 1 used to determine the actual number of columns and rows of the table, as well as algorithm 2 are used to determine the boundary line of the property. Tests conducted at 100 tabular HTML format, and the test results provide the accuracy of the algorithm 1 is 99.9\% and the accuracy of the algorithm 2 is 84\%.

Keyword: extraction algorithm, html, table extraction, table property

Introduction

The Internet provides the data in various model and formats. The table is one form that is used to display the data. Likewise with the format used such as HTML and PDF formats. The data in the table on the Internet can be retrieved by manual means that copy-paste, but it is an obstacle if done on more than one table, so in this article discussed about development an approach in preparation to find an area, so that the HTML format table extraction into a database can be done. Extraction or called wrapper that is part of the application that makes the web resource become a resource that can be queried since it is in the form of a database, where the source is in the form of semi-structured or unstructured. \cite{4}

Table considered as a representation of a structured data and related information in the form of 2 (two) dimensions. \cite{6}, the contents of the table are the data that are presented briefly. The table consists of the cell, which can contain labelling cell and data cell. \cite{7}.
Research on the extraction table been done by [7], which distinguishes the cell content by identifying the cell as property and the cell as an instance, and in this case the property is located in the first row of the table.

[8] Do the extraction of a table that presents biological data by 2 (two) ways: i). table detection (table identification of documents), and ii). Processing table (it extract data from tables). Detection of property has done by considering <hr> tag which is used to create a horizontal line in HTML. Document Object Model (DOM) is also used by researchers to perform the extraction of HTML tables, some of them [5], and [2]. DOM is a base or a stand-alone language used to represent and make the connection between objects of different documents into HTML or XML web page, with the form of a tree structure as depiction. [3] Combine DOM with XY cut algorithm to find a table on the web page.

Search schema matching on the extraction of HTML tables made to merge the results of extraction, but the research on this area is found several problems including the table location in web pages, merged attribute, and word synonyms in the process of merging the data extracted. [1]

Approach in preparation for finding areas (arrays) on the extraction of HTML tables in this article with respect to the position of property that could be more than 1 (one) rows in the table. (Figure 1)

![Figure 1 Examples of Table with Property More than 1 row](image)

Figure 1 shows the table property in row positions, which are row 1 and row 2. Property could be more than 1 (one) row because of the merging of rows and columns, as in Figure 1 shows a merging the 1st and 2nd column "Name" and the merging of row 1 and 2 "Phone Number". Then it shows the content / data table (instance) are from row 3 to row 7.

This study develop two algorithms which are preparation to find areas (arrays) to perform the extraction of HTML tables and show the results of the accuracy testing of those algorithms. Algorithm 1 is used to calculate the number of columns and rows, and the second algorithm is used to determine the row boundary of the table property.

The article is divided into four parts: the first part as an introduction that contains the definition of the problem and refer to previous similar research papers. The second part discuss the approach taken to perform the extraction of HTML table that contains 2 (two) algorithms, and the third part contain testing of the algorithm, and the final part contains the conclusions and future work.
2. Table Extraction Approach

A table has property and data cell. Refer to property and data cell the data can be considered in some types. Actually, in general there are three types of table. The first type is a table that has property in top row of table and the most left of table. The second type is a table that has property in top row of table. The third type is a table that has no property.

Refer to type of table, extraction process need to identify the area of property and cell. The first step is to calculate number of row and column in a table. The second step is to consider which row or column as property cell.

There are 2 (two) algorithm as a preparation in finding the area (array) to perform the extraction of HTML tables. Algorithm 1, that is algorithm used to calculate the actual number of columns and row. This is done since if the table has a property of more than 1 (one) row, then there is a merging of rows and columns in the property cell of the tables, so we need an algorithm to calculate the actual number of columns and row of the table. The calculation of the number of columns and row are actually useful to know the size of the table (row x columns).

**Algorithm 1 Count for Actual Number of Row and Actual Number of Column**

Read HTML
s = 0
If read <tr> then s = s + 1
RsTotal = s
Jum <td> = count tag <td>...</td> in first tag <tr>...</tr>
CsTotal = 0
For i = 1 to jum<td>
    Read nilai cs (i)
    CsTotal = CsTotal + cs (i)
Next i ;

Notes:
- RsTotal : the total number of <tr> tag on tag <table>...</table>
- CsTotal : the number of colspan value
- Jum<td> : the number of <td> tag
- cs : colspan value
- i : <td> tag
- s : <tr> tag

Figure 2 is an illustration that conducted to run the algorithm 1. In Figure 2 looks <HTML> tag of the example table in Figure 1 with the actual number of rows = 7, which is calculated by summing the <tr> tag inside the <table>... </table> tag. The actual number of columns = 3 is calculated by looking at the number of <td> tags inside the first <tr>... </tr> tag and if there is colspan in it, then it calculate the number of colspan.

Algorithm 2 is used to find the value of the row boundary as property (rowmax_pro) by finding the largest value of rowspan (RsMax) present in each ith tag <td> ... </td> on the sith tag <tr> ... </tr> . If it is not found rowspan value > 1 any longer, then the row boundary of the property is found. Figure 3 is an illustration to run the algorithm 2.
Figure 2. Illustration for Algorithm 1
Algorithm 2. Finding the Largest Rowspan Value, and Number of Row to be The Property Boundary

\[
\text{mBatas} (0) = 1 \\
\text{while } s = 1 \text{ do} \\
\quad \text{rsMax} (0) = 1 \\
\quad \text{Count jum}<\text{td}> \\
\quad \quad \text{For } i = 1 \text{ to } jum<\text{td}> \\
\quad \quad \quad \text{If } rs (i) > 1 \text{ then} \\
\quad \quad \quad \quad \text{If } rs (i) \geq rsMax (i-1) \text{ then } rsMax (i) = rs (i) \\
\quad \quad \quad \quad \text{else} \\
\quad \quad \quad \quad \quad \text{If } rs (i) < rsMax (i-1) \text{ then } rsMax (i) = rsMax (i-1) \\
\quad \quad \quad \text{Next } i ; \\
\quad \quad \text{mBatas} (s) = rsMax (i) + s - 1 \\
\quad \text{if } mBatas (s) < mBatas (s-1) \text{ then } mBatas (s) = mBatas (s-1) \\
\quad \text{until } mBatas (s) ; \\
\quad \text{rowmax_pro} = mBatas (s) ;
\]

Notes:
- RsMax : the highest value of rowspan
- mBatas : row value boundary as property
- rowmax_pro : row boundary which named as property
- i : \text{<td>} tag
- s : \text{<tr>} tag

Figure 3 shows an \text{<HTML>} tag from the example of table in Figure 1; they are the tag for row 1 up to row 3. \text{<td>} tag existing on the first \text{<tr>} \ldots \text{</tr>} tags have the largest rowspan value which is 2, so the row boundary of the table property exist in a table on the 2nd row, because once doing a reading for a \text{<td>} tag that is inside the second \text{<tr>} \ldots \text{</tr>} tags is no longer found any rowspan.

After obtaining the actual table size (row x columns) and the boundary row of the table property, then table extraction into a database is easy to do.

Illustration of implementation of the approach can be implemented in many areas. For instance in manufacturing area, the main issues is to find a raw material. Currently many suppliers provide the information in Internet by using table form. Automotive industry need to have spare parts, such as tire, engine audio system. There are many products of audio system that inform by table in Internet. To find the appropriate spare parts the manufacturing will copy many tables from many suppliers of audio system. The next step, the content of table will merge to be one table. This effort is acceptable for number of data and sources are limited. If number of data and sources are huge, it can be hard effort and difficult to avoid an error. By implementation the two algorithms, the automatic process data harvesting process in table of Internet can be performed.
3. Experiment

Both algorithms in this article are implemented using the PHP programming language. The test preparation is provide a table with various forms in HTML format totalling one hundred tables. The format of tables that used in experiment is as follow:

1. Basic form table consists of one or more row and columns, where the first row and the first column on the left of the property.
2. Basic form table consists of one or more row and columns, where 1 is the top line of the property, without any column as a property set.
3. Basic form table consists of one or more row and columns, where the tables are not property.
4. Basic form table consists of one or more row and columns, where the rows and columns of the property, and no merging row in the column property.
5. Basic form table consists of one or more row and columns, where the rows and columns of the property, and no merging columns on the property row.

![Figure 3 Illustration for Algorithm 2](image-url)
6. Basic form table consists of one or more rows and columns, where the rows and columns of the property, and no incorporation of rows and columns on the property row.

7. Basic form table consists of one or more rows and columns, where the top line of the property, there is no merging row and columns on the property row.

8. Basic form table consists of one or more rows and columns, where the table does not have a property but there was merging lines.

9. Basic form table consists of one or more rows and columns, where the table does not have a property but there was merging columns.

10. Basic form table consists of one or more rows and columns, where the table does not have a property but there was merging of rows and columns.

11. Basic form table consists of one or more rows and columns, where the tables have property on the position of row and there is a merger of both row and column of the merger on the property or instance. (Complex tables)

In Figure 4 shows one form of tables that are used as test inputs. The table in Figure 4 is a complex table which has property in first until third row with some row consists of span row.

| NO | FIRST NAME | MIDDLE NAME | LAST NAME | DATE OF BIRTH |
|----|------------|-------------|-----------|---------------|
| 1  | Angga      | Kurniawan   | Putra     | 01 02 91 n   |
| 2  | Dian       | Panaga      | Addya     | 02 02 92 n   |
| 3  | Cika       | Muziana     | Amira     | 03 06 93 n   |
| 4  | Dewi       | Aangwara    | Putri     | 04 07 n n n  |

Figure 4 Examples of Complex Table Form for Trial

The purpose of testing the algorithm 1 is to determine the ability of the algorithm to count the number of rows and number of columns of the table, while the second algorithm testing aims to determine the ability of the algorithm to find the limits of the table properties ranging from row 1 to a particular row number in the table. Here is the test scenario:

1. Algorithm 1 and algorithm 2 is written in the PHP programming language.
2. Input is HTML tables with different forms.
3. Output measured is accuracy of the actual number of rows and columns of the table that will be extracted (from algorithm 1), and the accuracy of determining the row boundary of the table property (from algorithm 2)

In Table 1 shows a summary of the test results of algorithm 1 to determine the actual number of rows and columns.
Table 1 Testing Summary of Algorithm 1

| Table Model | Manual Number of Row | Manual Number of Column | Program Number of Row | Program Number of Column | Accuracy (%) |
|-------------|----------------------|-------------------------|-----------------------|--------------------------|--------------|
| 1           | 4                    | 3                       | 4                     | 3                        | 100          |
| 2           | 5                    | 3                       | 5                     | 3                        | 100          |
| 3           | 5                    | 4                       | 5                     | 4                        | 100          |
| 4           | 6                    | 3                       | 6                     | 3                        | 100          |
| 5           | 6                    | 4                       | 6                     | 4                        | 100          |
| ...         | ...                  | ...                     | ...                   | ...                      | ...          |
| 78          | 5                    | 17                      | 4                     | 17                       | 90           |
| ...         | ...                  | ...                     | ...                   | ...                      | ...          |
| 98          | 13                   | 13                      | 13                    | 13                       | 100          |
| 99          | 13                   | 13                      | 13                    | 13                       | 100          |
| 100         | 16                   | 19                      | 16                    | 19                       | 100          |

Table 1 shows the comparison of the number of columns and number of rows retrieved manually (human visual) and the number of rows and columns obtained from algorithm 1 with the program. The average percentage of the number of rows and columns accuracy of 100 table test is 99.9%.

Then Table 2 shows a summary trial in algorithm 2 to determine the accuracy of the table to determine the property boundaries on what row.

Table 2 Testing Summary of Algorithm 2

| Table Model | Number of Row to be property Boundary | Accuracy (%) |
|-------------|-------------------------------------|--------------|
|             | Manual Number of Row | Program Number of Row |               |
| 1           | 1                    | 1                        | 100          |
| 2           | 1                    | 1                        | 100          |
| 3           | 1                    | 1                        | 100          |
| ...         | ...                  | ...                      | ...          |
| 59          | 1                    | 2                        | 50           |
| 60          | 3                    | 2                        | 66.7         |
| ...         | ...                  | ...                      | ...          |
| 99          | 3                    | 3                        | 100          |
| 100         | 4                    | 4                        | 100          |

Table 2 shows there are a tables that has the accuracy is not 100% (e.g., in models of tables 59 and 60). The reason of the result because the tables are complex model. Refer to the result if neglected the tables achievement is 100%, however the achievement is 99% if include table 59 and 60.

Refer to the result of algorithm 2, the approach has limitation in simple table to provide the best result.
4. Conclusion

The experiment out for the algorithm 1 and algorithm 2, it was found that the percentage of accuracy algorithm 1 to get the actual number of rows and columns of the table is 99.9%, the result is better then Tangli [7] research with achievement 91%.

Percentage accuracy of the algorithm 2 to determine the table properties row boundary is 84%. Result of algorithm is enhance achievement compare to Wong [8] result (83%).

Looking at the percentage of accuracy of the two algorithms, it is said that the approach taken to prepare the area (array) as the beginning of the process of extracting HTML tables work well.

In a subsequent study, it will use the algorithm 1 and algorithm 2 to conduct the cell contents of the property, and making the contents of cell as an instance in the table extraction process.

References

[1] Embley D. W., Hurst M., Lopresti D., Nagy G. 2006 Table-Processing Paradigms : a Research Survey International Journal of Document Analysis, page 66-86

[2] Gultom R., Sari R. F., Budiarjo B. 2011 Proposing the new Algorithm and Technique Development for Integrating Web Table Extraction and Building a Mashup Journal of Computer Science 7 (2), page 129-142

[3] Krupl B., Herzog M., Gatterbauer W. 2005 Using Visual Cues for Extraction of Tabular Data from Arbitrary HTML Documents Proceeding WWW '05 Special interest tracks and posters of the 14th international conference on World Wide Web

[4] Lerman K., Knoblock C., Steven M. 2001 Automatic Data Extraction from Lists and Tables in Web Sources Proceedings of Automatic Text Extraction and Mining Workshop ([ATEM]-01)

[5] Lin J., Wong J., Nichols J., Cypher A., Lau T. A.2009 End-User Programming of Mashups with Vegemite Proceedings of The T3th International Conference on Intelligent User Interfaces, page 97-106

[6] Liu Y., Mitra P., Giles C.L. 2008 A Fast Preprocessing Method for Table Boundary Detection : Narrowing Down the Sparse Lines using Solely Coordinate Information The Eighth IAPR International Workshop on Document Analysis Systems, page 431 – 438

[7] Tengli A., Yang Y., Ma N. L. 2004 Learning Table Extraction from Examples Proceedings of The 20th International Conference on Computational Linguistics

[8] Wong W., Martinez D., Cayedon L. 2009 Extraction of Named Entities from Tables in Gene Mutation Literature Proceedings of The Workshop on Current Trends in Biomedical Natural Language Processing, BioNLP '09