Editing Practice

DIGITAL PUBLISHING

Application research on table structure recognition and information extraction in sci-tech academic journals based on visual studio tools for Office technology

Lipeng Wang, Jie Chen*, Chunyu Zheng, Jie Feng

China Medical University Journal Center, Shenyang 110122, Liaoning Province, China

ABSTRACT

The premise of intelligent table processing in Word is to extract the table structure and text information. By using visual studio tools for Office (VSTO) to obtain the extensible markup language (XML) information of the table, the structural relationship of the table and the text format of each cell can be further recognized. Compared with Visual Basic for Applications (VBA) technology, VSTO technology is slower in handling Word, but it has better extensibility and efficiency than VBA. VSTO technology can effectively recognize the structure of the table and extract information, providing possibilities for subsequent intelligent processing.

Key words: table, visual studio tools for Office, journal, editor

INTRODUCTION

Tables are common representations in documents, with the compact and easy-to-understand format that allows for faster retrieval and comparison of information. They are widely used in science and technology (hereinafter abbreviated as sci-tech) academic journal papers and they are important data objects within documents. Identifying tables can assist with information input, reducing time and labor costs. In fields such as scientific research and statistics, the extracted structured data from table recognition results have wide applications, such as query and response systems,[1] construction of scientific research rankings,[2] and extraction of biological experiment features,[3] *etc*. When editing and proofreading sci-tech academic journals, editors often need to modify the tables, such as flipping tables,[4] making three-line tables,[5] and verifying table data.[6] These tasks consumed a lot of time and inevitably led to various editorial errors. Intelligent processing of tables can not only reduce a large amount of repetitive work but also improve the editing and proofreading quality of journals. The first step in intelligent table processing is to extract the structure and text format information. The tables have various styles of structures and different text formats in sci-tech academic journals. In recent years, parsing document tables and performing table recognition and extraction has become a research hot spot. Scholars have explored the use of optical character recognition (OCR) for document page image table detection and recognition. According to different logical inputs, non-structured images are located, tables are detected, the logical structure of each cell is further reconstructed, and the text content of each cell is identified. Finally, structured table data is output.[7–10] However, previous research has mainly focused on image-based documents and PDF documents, and there has been little research on the automation processing of editable text in sci-tech academic journal editing and proofreading work. The logical modes used are diverse and of limited value for reference. Wang *et al.*[6] used

***Corresponding Author:**
Jie Chen, China Medical University Journal Center, No.77, Puhe Road, Shenyang North New Area, Shenyang 110122, Liaoning Province, China. Email: chenjiesunyt@163.com; https://orcid.org/0000-0002-8206-0255

Received: 5 June 2023; Revised: 26 June 2023; Accepted: 20 July 2023; Published: 31 July 2023
https://doi.org/10.54844/ep.2023.0412

visual basic for applications (VBA) to extract information from tables in Word, but it had certain limitations in practical work. This paper explores to use visual studio tools for Office (VSTO) technology to achieve recognition of tables' structure and text format information.

WHY SELECT VSTO TECHNOLOGY

VBA technology is more convenient for simple Word automation operations, especially since Word itself provides macro recording functions that can be modified to speed up development. However, VBA is still a scripting language, and its execution depends on the parent application program, such as Excel, Word, *etc.* At the same time, its extension has limitations,[11] and it belongs to a document-level application, which poses certain security issues. Currently, the rapid development of artificial intelligence (AI) proofreading and the emergence of various natural language processing (NLP) models make VSTO, as a program-level application development, more advantageous.

TO OBTAIN DOCUMENT OBJECTS

In VSTO, you can create a project using Word add-ins to operate documents by loading them as Word add-ins,[12] or you can create a general form application and reference the "Microsoft.Office.Interop.Word" assembly to create and open Word documents within the form application. Both methods applications can obtain the "Microsoft.Office.Interop.Word.Document" object (the variable "doc" represents this object throughout the code), which can be used to perform various operations on the document.

TO OBTAIN TABLE OBJECTS

Executing the code "Tables tables = doc.Range().Tables;", the program will obtain the collection of tables in the document, which is saved in the collection variable "tables". Specific table objects can be obtained using their index. In VSTO, the numbering of document element objects starts from "1" instead of "0", so the first table object is "tables [1]". The variable "table" will be used to represent the operation table object.

TABLE STRUCTURE RECOGNITION

Issues with table structure recognition

For a regular table without merged cells, the information of each cell can be obtained and the table structure can be extracted by iterating through rows and columns using the "Cell (int Row, int Column)" method of the table. However, when dealing with tables that contain merged cells, this method will prompt error messages.[13] Lin proposed a solution to first obtain the object of the first cell in the table "table.Cell(1,1)", and then traverse all the cells through the "Next" property of the cell object, recording the row index "RowIndex" and column index "ColumnIndex" of each cell, and extracting the column width information of each cell.[13] The table structure can be obtained by designing an algorithm. This method frequently accesses Word element information and is suitable for VBA development, but it runs less efficiently for desktop programs that use communication with the Word process to obtain information.

The specific implementation of table recognition

Starting from Word 2007, Word saves documents in the OpenXML format with a file extension of ".docx". The document structure is described using extensible markup language (XML). The XML code of a table can be obtained by using the code "table.Range.WordOpenXML". By analyzing the XML code, the table structure and text formatting information can be obtained. OpenXML is an XML-based document format specification proposed by Microsoft, which became an open international standard and is applied starting from Word 2007.[14] OpenXML software development kit (SDK) is a collection of classes that can create and process documents following the OpenXML document format.[15] It can be imported through the NuGet package manager in the Visual Studio development environment. The development version is written in C# language, and developers can use C# to generate various safe and reliable running programs in .NET.[16]

Nesting relationship of table XML elements

Base on this framework, the "w:tbl" element represents a table, the "w:tr" element represents a row in the table, and the "w:tc" element represents a cell in the table. The nesting relationships of these elements are shown in Figure 1.

```
- <w:tbl>
    + <w:tblPr>
    + <w:tblGrid>
    - <w:tr w:rsidTr="00A33BB6" w:rsidR="0075031B">
        + <w:tc>
        + <w:tc>
      </w:tr>
    + <w:tr w:rsidTr="009677E7" w:rsidR="0075031B">
    + <w:tr w:rsidTr="0075031B" w:rsidR="0075031B">
  </w:tbl>
```

Figure 1. The nesting relationship of the extensible markup language document structure for tables.

Extraction of table cells' merging information

The coding using VSTO technology can extract information regarding merged cells in a table. As shown

in Figure 2, the table has merged cells both horizontally and vertically. The XML code snippet for this table is shown in Figure 3. If the "tc" element contains a "gridSpan" node, it indicates that there are merged cells in the row direction, with the "val" attribute value indicating how many cells are merged from the current cell (including the current cell). If the "tc" element contains a "VMerge" node, it indicates that there is a merging in the column direction, and if the "val" attribute is "restart", it indicates the starting cell. Based on this information, the merging information for the entire table can be extracted. The pseudocode for extracting merged cell information from tables is shown in Figure 4.

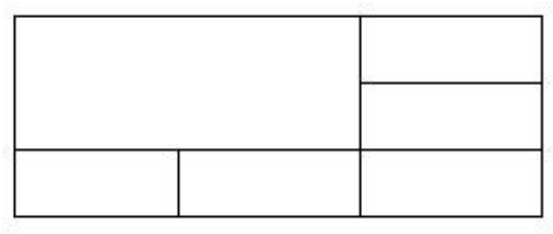

Figure 2. Table with merged cells in rows and columns.

```
- <w:tbl>
    + <w:tblPr>
    + <w:tblGrid>
    - <w:tr w:rsidTr="00A33BB6" w:rsidR="0075031B">     (1)
        - <w:tc>     (2)
            - <w:tcPr>
                <w:tcW w:w="5681" w:type="dxa"/>
                <w:gridSpan w:val="2"/>     (3)
                <w:vMerge w:val="restart"/>     (4)
              </w:tcPr>
              <w:p w:rsidR="0075031B" w:rsidRDefault="0075031B"/>
          </w:tc>
        + <w:tc>
      </w:tr>
    - <w:tr w:rsidTr="009677E7" w:rsidR="0075031B">     (5)
        - <w:tc>     (6)
            - <w:tcPr>
                <w:tcW w:w="5681" w:type="dxa"/>
                <w:gridSpan w:val="2"/>
                <w:vMerge/>     (7)
              </w:tcPr>
              <w:p w:rsidR="0075031B" w:rsidRDefault="0075031B"/>
          </w:tc>
        + <w:tc>
      </w:tr>
```

Figure 3. extensible markup language code snippet for a table with merged cells. (1) The first row. (2) The first cell in the first row. (3) Merge two cells in the row direction. (4) Merge cells in a column direction starting from the cell. (5) The second row. (6) The first cell in the second row. (7) This cell has already been merged in the column direction.

Extraction of cell text format information

The text in a cell is saved in the "w:r" element. If the "r" element contains a node with the attribute value of "superscript", then the text is represented as superscript. If the attribute value is "subscript", then the text is represented as subscript. If the "r" element contains a "w:i" or "w:iCs" element, then the text is represented as italic. The specific information for the text is stored in the "t" element, as shown in Figure 5. For more information on other formatting options, please refer to the Microsoft official documentation.

DISCUSSION

The significance and existing problems of table processing

Tables can be divided into two categories according to their functions: information processing tables and mathematical calculation tables.[17] Both types of tables can be found in sci-tech academic journals. Currently, the standardization of Chinese sci-tech academic journal tables requires that they should be self-explanatory, with a focus on simplicity of structure, and logical subject-predicate relationships between horizontal and vertical labels.[18] The subject is on the left side of the table and the predicate is on the right side of the table. Generally, the format uses three-line tables.[18] In the current editing work, there are a large number of non-standard tables that need to be manually processed, which causes a lot of difficulties for editors. Errors are inevitable in the manual adjustment process, making the urgent need for automated processing of editable text tables. The current researches on table detection and recognition mainly focus on image documents or portable document format (PDF) documents, using OCR technology based on heuristic rules[19] or deep learning methods such as Faster Region-based Convolutional Neural Network (R-CNN),[20] You Only Look Once version 3 (YOLOv3),[21] Fully Convolutional Networks (FCN),[22] Graph Convolutional Networks (GCN), and Deformable Convolutions[23] to detect and recognize tables that meet specific conditions. Although these recognition models have achieved good results, their research achievements are far from the editing work. For OCR technology recognition of editable text in sci-tech academic journals editing and proofreading, additional conversion into an image or PDF document is required for processing, and the accuracy is poor, so its usefulness is limited. How to use existing NLP technology to solve practical problems in editing work is one of the more practical directions in editing studies. Wang *et al.*[6] used VBA technology to explore information extraction from tables. The results showed feasibility, but the tables in that literature required a simple and recognizable structure or needed to undergo editing and processing. This idealized state is far from the actual workload of sci-tech academic journal editors. In this study, we included natural source manuscripts that were not edited or processed by editors. Based on the inherent logic of the tables, we transformed them into software logic using VSTO technology for research and exploration, and the results showed that automated processing was feasible and effective. Therefore, we believe that VSTO technology can effectively identify table structures and extract

```
public static TableInfo GetTableInfo(Table oldTalbe)//oldTalbe:the table from which information needs to be extracted
 {
     String xml = oldTalbe.Range.WordOpenXML;//get the XML code of the current table
     MatchCollection rowXmls = Regex.Matches(xml, @"\<w:tr.+?\</w:tr\>");//get a collection of XML codes for table rows
     TableInfo  tableInfo = new tableInfo(oldTalbe.Rows.Count + 1, oldTalbe.Columns.Count + 1); //object to save table information
     int rowNumber = 0; //row counter
     foreach (Match rowXml in rowXmls)//iterate through each row of the original table
     {
         rowNumber++;
         tableInfo.rows[rowNumber] = new LvyeRow(oldTalbe.Columns.Count + 1);
         MatchCollection cellXmls = Regex.Matches(rowXml.Value, @"\<w:tc\>.+?\</w:tc\>");//get a collection of XML codes for cells
         int cellNumber = 0;//cell counter
         foreach (Match cellxml in cellXmls)//iterate through each cell of the row
         {
             //get column merge information
             String vMergeRestart = Regex.Match(cellxml.Value, @"w:vMerge w:val=""restart""").Value;
             Boolean isvMergeRestart = false;
             Boolean isvMerge = false;
             if (vMergeRestart != "")//if it is the start of a column merge
             {
                 isvMergeRestart = true;
             }
             else
             {
                 String vMerge = Regex.Match(cellxml.Value, @"w:vMerge").Value;
                 if (vMerge != "")//if it is in a merged state in the column direction
                 {
                     isvMerge = true;
                 }
             }
             //get row merge information
             String gridSpan = Regex.Match(cellxml.Value, @"(?<=\<w:gridSpan w:val="")\d+").Value;
             if (gridSpan != "")//if it is a row merge
             {
                 int spanCount = int.Parse(gridSpan);
                 cellNumber++;
                 //save the cell information
                 AddCell(tableInfo, rowNumber, cellNumber, spanCount, isvMergeRestart, isvMerge, cellxml.Value);
                 for (int i = 1; i < spanCount; i++)
                 {
                     cellNumber++;
                     AddCell(tableInfo, rowNumber, cellNumber, 0, isvMergeRestart, isvMerge, "");
                 }

             }
             else
             {
                 cellNumber++;
                 AddCell(tableInfo, rowNumber, cellNumber, 0, isvMergeRestart, isvMerge, cellxml.Value);
             }

         }//cell iteration ends
     }//row iteration ends
     return tableInfo;
 }
```

Figure 4. The pseudocode for extracting merged cell information from tables.

Superscript

```
- <w:r w:rsidRPr="00285E14">
    - <w:rPr>
        <w:rFonts w:hint="eastAsia"/>
        <w:vertAlign w:val="superscript"/>
      </w:rPr>
      <w:t>2</w:t>
  </w:r>
```

Subscript

```
- <w:r w:rsidRPr="00285E14">
    - <w:rPr>
        <w:rFonts w:hint="eastAsia"/>
        <w:vertAlign w:val="subscript"/>
      </w:rPr>
      <w:t>2</w:t>
  </w:r>
```

Italic

```
- <w:r w:rsidRPr="00285E14">
    - <w:rPr>
        <w:i/>
      </w:rPr>
      <w:t>P</w:t>
  </w:r>
```

Figure 5. Text format extensible markup language code snippet.

information, providing ideas that can be useful for the automation and intelligence of table processing, as well as representing a valuable exploration.

Table processing based on VSTO technology

In this study, we used VSTO technology to obtain the XML code of editable text tables, customized the coding program, and conducted table parsing to achieve automatic recognition and information extraction. OpenXML is an international open standard for word processing documents, presentations, and spreadsheets, which can be implemented free of charge by multiple applications on multiple platforms and is widely used in editable documents. Shen explored the use of OpenXML technology to batch process ".docx" file format checks in Word documents and applied them to the inspection of the document format of our school's graduation thesis, achieving good results.[24] Guo designed a file review system based on OpenXML using a browser/server (B/S) architecture, exploring the implementation of batch processing, automated format

review, format repair, document desensitization, task assessment, and level evaluation, and achieved preliminary results.[25] Currently, there are many domestic technical developments based on OpenXML, which are mostly used for automated format review of documents, encrypted file desensitization processing, *etc.*, while for sci-tech academic journal table processing, improved VBA technology is often used. VBA technology can achieve one-click processing of tables, but its accuracy and refinement are insufficient.[26–28] We found that there is currently no report on the application of VSTO technology to automatic processing of sci-tech academic journal tables. VSTO technology has strong recognition capabilities for complex element-level cells, can achieve one-click modification, and can be personalized, meeting the basic requirements of sci-tech academic journals for table editing. In this study, we used VSTO technology based on OpenXML to create, modify, and process Chinese sci-tech academic journal tables, using specific markings and element recognition. In editable text documents such as Word, tables can be represented by the <w:tbl> element, which can contain the <w:tr> element. The <w:tr> element can contain the <w:tc> element, and the <w:tc> element can contain the <w:p> element, which can contain text or other elements. By using these elements, various complex tables were created and coded to automatically recognize and set styles. At the same time, the tables were also automatically modified and processed, achieving good results.

Inherent requirements of Chinese sci-tech academic journals for automated processing

Chinese sci-tech academic journals have the characteristics of fine disciplinary classification, large quantity, diverse writing teams, specific readership groups, and unique content and expression forms. Their content covers hundreds of subjects and cross-disciplinary fields in 12 categories. For example, China National Knowledge Infrastructure (CNKI) has collected more than 6500 Chinese sci-tech academic journals in 105 disciplines.[29] However, facing massive data, intelligent development is restricted by the lack of basic technological foundation, resulting in limited output. The bottleneck problems include industry attention, interdisciplinary talents, and technical barriers. The issue related to technology is how to design an effective and convenient application system to gradually overcome the difficulties. Currently, the application of software development in the field of sci-tech academic journals in China is still at an early exploration stage and is not yet mature. The application direction, foundation, mode, results, and technical feasibility all need further exploration. Zheng *et al.*[30] believe that the current flawed standard system, lack of standardization execution, and non-uniformity of standards in the domestic sci-tech academic journals are the main factors inhibiting the development of artificial intelligence, therefore, starting from the top-level design of the system, ample sample collection and research should be conducted, and existing national standards should be integrated and updated to formulate sci-tech academic journal national standards suitable for the development of software assistants and promote their specific execution standards. Tables are one of the components of sci-tech academic journal manuscripts, with relatively uniform presentation forms and writing styles, good standardization execution, and neat and standardized formatted content. Therefore, it is feasible to automatically process them at the technical level.

CONCLUSION

The structure of tables and information extraction are the preparatory work for intelligent processing of tables. This article proposes a solution for extracting structure and text format information from common tables in sci-tech academic journals using VSTO technology, providing foundational guarantees for subsequent intelligent table operations.

DECLARATIONS

Author contributions

Wang LP: Writing—Original draft. Chen J: Project administration, Writing—Review and Editing. Zheng CH: Resources, Conceptualization. Feng J: Inductive documents.

Source of funding

This work is supported by the 2023 Liaoning Provincial Natural Science Foundation Project (General Project) (No.2023-MS-146) and the 2023 Natural Science Journal Editing Research Association of the Chinese Academy of Sciences (No.YJH202319).

Conflict of interest

The author has no conflicts of interest to declare.

REFERENCES

1. Park C, Kim M, Park S, Lim S, Lee J, Lee C. Korean tableQA: Structured data question answering based on span prediction style with S-3-NET. *ETRI J*. 2020;42(6):899–911.
2. Hou YF, Jochim C, Gleize M, Bonin F, Ganguly D. Identification of tasks, datasets, evaluation metrics, and numeric scores for scientific leaderboards construction. In: *Proceedings of the 57th annual meeting of the association for computational linguistics*. Association for Computational Linguistics; 2019: 5203–5213.
3. Milosevic N, Gregson C, Hernandez R, Nenadic G. A framework for information extraction from tables in biomedical literature. *Int J Doc Anal Recognit*. 2019;22(1):55–78.
4. Wang CD. [Inversion of Subject and Predicate in Tables of Scientific Papers and Its Revision]. *Chin J Sci Tech Period*. 2008;19(4):680–682.

5. Yu Y. [Correction of a three-line table with multiple errors]. *Acta Editol.* 2021;33(3):348.
6. Wang JX, Kang LY, Li YY. [Promotion of table editing quality based on Word VBA]. *Acta Editol.* 2021;3(3):322–326.
7. Zucker A, Belkada Y, Vu H, Nguyen VN. ClusTi: Clustering method for table structure recognition in scanned images. *Mob Netw Appl.* 2021;26(4):1765–1776.
8. Zhu YY, Yao C, Bai X. Scene text detection and recognition: Recent advances and future trends. *Front Comput Sci.* 2016;10(1):19–36.
9. Long SB, He X, Yao C. Scene text detection and recognition: The deep learning era. *Int J Comput Vis.* 2021;129(1):161–184.
10. Zhang MZ. [Design and implementation of a table recognition system based on deep learning]. Wuhan: Huazhong University of Science and Technology. 2020. (Thesis)
11. Liu YF. [*Intermediate-level tutorial on VSTO development*]. Tsinghua University Press; 2019: 218.
12. Liu YF. [*Beginner's tutorial on VSTO development*]. Tsinghua University Press; 2017: 134.
13. Lin Y. [Method for reading cell attribute values in Word tables using programming language]. *Comput Program Skills Maint.* 2021(6):31–32+52.
14. About the Open XML SDK 2.5 for Office. Microsoft. Accessed June 1, 2023. https://learn.microsoft.com/en-us/office/open-xml/about-the-open-xml-sdk
15. What's new in the Open XML SDK 2.5 for Office. Microsoft. Accessed June 1, 2023. https://learn.microsoft.com/en-us/office/open-xml/what-s-new-in-the-open-xml-sdk
16. A tour of the C# language. Microsoft. Accessed June 1, 2023. https://learn.microsoft.com/en-us/dotnet/csharp/tour-of-csharp/
17. Introduction. In: Campbell-Kelly M, Croarken M, Flood R, Robson E, eds. *The history of mathematical tables*. Oxford University Press; 2007: 3–5.
18. National Press and Publication Administration. [Specification of academic publishing—Table (CY/T 170—2019)]. Accessed June 1, 2023. https://journal.scu.edu.cn/info/1253/1221.htm
19. Koci E, Thiele M, Romero O, Lehner W. A genetic-based search for adaptive table recognition in spreadsheets. In: *2019 International Conference on Document Analysis and Recognition*. IEEE; 2020: 1274–1279.
20. Ren SQ, He KM, Girshick R, Sun J. Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Trans Pattern Anal Mach Intell.* 2017;39(6):1137–1149.
21. Redmon J, Farhadi A. YOLOv3: An Incremental Improvement. arXiv. Accessed June 1, 2023. https://arxiv.org/abs/1804.02767
22. Shelhamer E, Long J, Darrell T. Fully convolutional networks for semantic segmentation. *IEEE Trans Pattern Anal Mach Intell.* 2017;39(4):640–651.
23. Dai JF, Qi HZ, Xiong YW, *et al.* Deformable convolutional networks. In: *2017 IEEE International Conference on Computer Vision*. IEEE; 2017: 764–773.
24. Shen LL. [Study on document format checking technology based on OpenXML]. *Electron Tech.* 2021;50(4):44–45.
25. Guo XQ. [Design and implementation of the review system for electronic design document]. Xi'an: Xidian University. 2019. (Thesis)
26. Mao X, Li YN, Dong L. [Automatic Editing and Proofreading of Scientific Papers Based on Word VBA]. *Tianjin Sci Tech.* 2020,47(1):98–101.
27. Wang ZX, Wang LM. [Applying VBA to batch extract and organize document data]. *Electron Tech Softw Eng.* 2019;15:154–155.
28. Yang QK. [Application of Word VBA in editing work]. *J News Res.* 2022;13(10):207–209.
29. Zhou WH, Chen DD. [China publishing integration development report in 2021]. *Sci-Tech Publ.* 2021;(5):60–69.
30. Zheng CY, Wang LP, Chen J. [Current standards for China scientific journals and related international progress]. *Chin J Sci Tech Period.* 2021;32(7):851–858.