Efficient XML Document Compressing Method Based on Internet of Things

Lv, Jiajia[a] , Wang, Yuanli[b], Zhong, Yi

State Engineering Laboratory of Fiber Optic Sensing Technology

Wuhan University of Technology

Wuhan, China, 430070

[a]345040854@qq.com, [b]wyl563@163.com

Keywords: Internet of Things; XML data; information redundancy; compression; XCiot

Abstract.XML data have been widely used in the Internet of Things. However, there are some problems that XML is of huge massive data and high information redundancy. Currently, XML document compression methods proposed by many researchers only focusing on compressing one single document, they have not considered that XML data are time-related redundant information in the IOT. Therefore, these compression methods are less effective for real-time data of XML in the IOT. According to the recurring feature of XML's contents and node path in the IOT, we propose a new efficient compressing method for real-time data of XML, called XCiot (XML Document Compressing Method Based on Internet of Things).Theoretical analysis and experimental results show that XCiot can largely improve the data compression ratio of real-time data of XML in the IOT. Thus, it can improve the transmission performance.

Introduction

On one hand, to achieve the information exchange and sharing between objects and objects as well as objects and human, and to make the intelligent management of network come true finally, the Internet of things has been developed. On the other hand, the XML (Extensible Markup Language) is gradually becoming the factual standard for cross-platform data representation and transmission, XML is often used in information transmission in the Internet of things. However, due to the self-describing and semi-structured character of XML, there is great data redundancy, for which the Internet of things transmission rate is severely affected. Thus, it becomes the new topic of Internet of things development that how to reduce data redundancy, make valid data compression of XML and improve network transmission ability.

Currently, one single XML file compression is often considered during the data compression in Internet of things. For the massive prior knowledge of XML real-time data [1], satisfying compression effect only achieved in single file but not in tons of XML real-time data, such as XMill [2], WINRAR, XMLPPM, etc. XCiot, here we proposed, takes full advantage of the similarity of XML message to do the compression. The result suggested that XCiot has a better compression ratio than traditional methods on Internet of things XML compression which is of high apriority.

Related Work

At present, there are too many compression algorithms at home and abroad, different algorithms show inconsistent compression efficiency on the same type data, some are only aimed at particular categories. It is usually considered that there are two kinds of XML file compressor, one is universal, and the other is XML dedicated.

By using traditional technology, the universal compressors treat XML documents as usual plain text documents when do the compression rather than optimize the process according to the structural characteristics of XML itself. The universal compressors cover WINRAR, gzip, bzip2, WinZip[3], PPM and so on, they show some advantages in common compression, compression time,

mature technique, but lower compression ratio on XML than the dedicated.

XML dedicated compressors will analyze XML documents and then take specially optimized ways on the basis of their structure and content, dedicated compressors show better compression ratio than universal compressors, but make process more time consuming. XML dedicated compressors include XMill, XMLPPM, SCMPPM, XAUST, rngzip, XGrind etc [4]. XMill is the first compressor among them, according to which the content information is separated from the structure data and the same categories are kept in one space, thus, higher compression ratio is achieved by using the same methods to process on, which make a great contribution on innovating the dedicated compressors.

On hand, some compression algorithms keep high redundant information from the XML documents to decrease the time and simplify the processes, on the other hand, many compressors, such as universal compressors, failed to make special processes on XML documents and remain the compressed files high redundant, so the compression ratio is to be improved.

In recent years, various improved compression algorithms are also springing up, some take the redundancy among documents into consideration when they process on multiply documents at the same time. In [5], Liang Li has come up with the idea that compression can be acquired by making use of the redundancy among documents. In [6], Moore, J.P.T. et al proposed a method that compression is based on a simple mapping of data values belonging to a set of data types to a series of integer values. In [7], Dongping Wei et al proposed a new Compressor Structural Join Oriented XML Data Compressor which makes structural join possible by giving all elements and attribute names in document a unique region encoding. In [8], Mansour, A.M.A. et al proposed an enhancement of the two levels dictionary based compression. In [9], Xiong Tao et al proposed a multi-modal structured dictionary learning algorithm. These methods provide some very important ideas to the XCiot in this paper.

XML Document Compressing Method Based on Internet of Things

A. The new Model of Internet of things

In recent years, in order to unify protocol specification and resource sharing, Internet of things have gradually developed from the original small-scale system to large-scale integration system [10]. This integrated Internet of things framework is shown in Fig. 1.

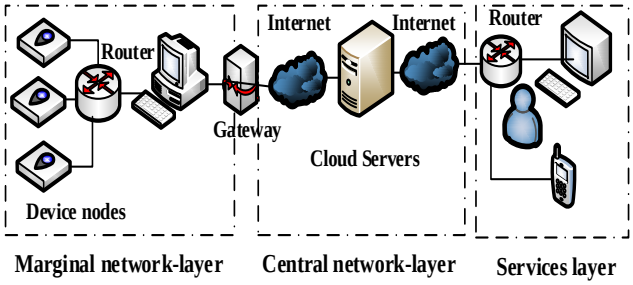

Figure 1. Integrated Internet of things framework

The integrated Internet of Things framework is divided into three layers: Marginal network-layer, Central network-layer and Services layer. Marginal network-layer is responsible for collecting various types of device node's data, and transform them into a standard data format, such as XML, PML, etc. A plurality of Marginal network-layer connected Central network-layer through the gateway and send the XML data to the cloud server of Central network-layer for data processing. After that, the Central network-layer sends processed data to the Services layer, and ultimately providing services to users.

B. Compression principle

The XCiot is applicable to IOT, where the special data source and the transport structures provide a runtime environment for XCiot. The IOT's Marginal network-layer corresponds to the data compression side and Central network-layer corresponds to the data decompression side. XCiot

has two main levels of compression. The first level is mainly for the dictionary compression of real-time data of XML's time-related redundant information. The second level is mainly for the compression of the XML's content and structure redundancy. The compression model of XCiot is shown in Fig. 2.

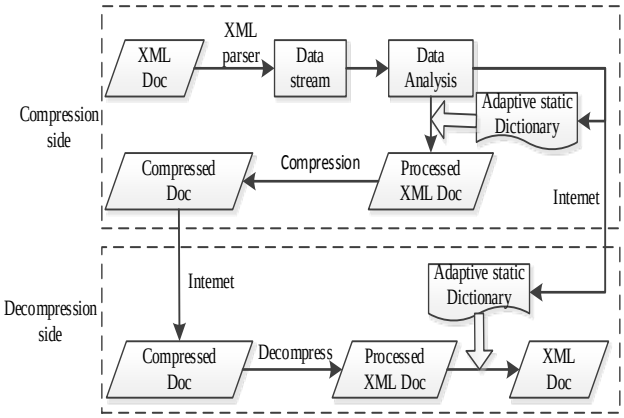

Figure 2. XCiot compression model

1) The specific steps are as follows:

Parse original XML documents and obtain their data information.

Collect data information of XML documents for statistical analysis and processing, establish the adaptive static dictionary by the algorithm of statistical pattern recognition, in which the shorter code reflects the high- probability string data, while the longer code reflects the low-probability one according to Huffman encoding theory.

The adaptive static dictionary is delivered from the compression side to the decompression side.

Realize the dictionary compression of the XML document by making use of the generated dictionary on the step (b), and produce the pre-compressed file of XML.

Parse the pre-compressed file of XML and separate its structure information and data information.

Compress the structure information and data information of the pre-compressed files of XML respectively, and generate the final compressed files.

The final compressed files will be delivered from the Marginal network-layer to the Central network-layer.

The Central network-layer will pre-compress the received compressed files and then execute a dictionary-decompression through static dictionary, which can recover the original data of XML.

Parse is very important in the compression process. In order to parsing XML documents faster, we use the DOM parsing mode, which regard the entire XML document as an object and the entire XML document is loaded into memory at first.

2) Example:

A standard XML document as shown in Example 1:

Example1: a standard XML document

```
<PLS>
    <userID>testUser</userID>
    <keyParameter>
                <parm>item1</parm>
                <parm>item2</parm>
                <parm>item3</parm>
                <parm>item4</parm>
                <parm>item5</parm>
                <parm>item6</parm>
                <parm>item7</parm>
                <parm>item8</parm>
                <parm>item9</parm>
    </keyParameter>
```

</PLS>

To better illustrate the process of generating a static dictionary, we mentioned a simple example. For example, after parsing, XML data information and its frequency during the process of generating a static dictionary are shown in the first two columns of Table I. According to the algorithm of statistical pattern recognition and data, we can generate a static dictionary, the third column of Table I correspond to the corresponding mapping.

TABLE I. STATIC DICTIONARY

Data information	Frequency	Code
testUser	20	1
item1	10	4
item2	10	5
item3	5	9
item4	6	8
Item5	8	6
Item6	3	10
Item7	19	2
Item8	12	3
Item9	7	7

We can obtain a pre-compressed document from the dictionary compression of an original XML document. If there is a new data information (newItem) not existing in the dictionary, we use the quotation marks it, as shown in Example 2:

Example2: a pre-compressed XML document

```
<PLS>
    <userID>1</userID>
    <keyParameter>
                <parm>4</parm>
                <parm>5</parm>
                <parm>9</parm>
                <parm>8</parm>
                <parm>6</parm>
                <parm>10</parm>
                <parm>2</parm>
                <parm>3</parm>
                <parm>7</parm>
                    <parm>’newItem’</parm>
    </keyParameter>
</PLS>
```

XCiot can express it with the shorter code by the comparative transition between the real-time data of XML and adaptive static dictionary, and further compress it. As in the process of compression, it is not necessary for the compression side to deliver dictionary to the decompression side every time, which thus greatly improves the compression ratio. As some new data in the XML can’t be matched to the adaptive static dictionary of dictionary compression by the compression side, therefore the content doesn’t have the dictionary compression and mark them. In the process of data analysis, record and analyze the new character string information. On the other hand, the decompression side will skip the dictionary compression, if they do not find the new information exist in the adaptive static dictionary.

During the compression of the pre-compressed XML document, we allocate the separated of structure and information to different containers, as it is shown in Table II:

TABLE II. DATA CONTAINERS

Structure table

1	/PLS
2	/PLS/userID
3	/PLS/ keyParameter
4	/PLS/ keyParameter/parm

/PLS/userID

1

/PLS/ keyParameter/parm

4	5	9	8
6	10	2	3
7	‘newItem’		

The values of each unique path (attribute) are stored in a separate table (container) by means of the idea of XMill. Therefore, the value of each container have become homogeneous, which will be more efficient for compression.

Adaptive static dictionary

To make the static dictionary adaptable for the changing data in the Internet of Things, we should let the dictionary have self-adaption and be able to dynamic update, the specific method is as follows:

We should keep collecting the real-time data of XML in the Internet of Things, analyze all XML data in every node and then record it in the statistical table.

According to the occurred probability of content in the statistical table, express the longer character string information with shorter code by application of statistical pattern recognition algorithm.

Analyze the difference between the new generated dictionary and the old version one. Mark new version number for the new dictionary, if there is a big difference between them.

New dictionary will be delivered from the compression side to the decompression side.

After the compression and decompression of new dictionary by compression side and decompression side, the static dictionary can be able to update and have self-adaptability.

The algorithm of statistical pattern recognition used in the paper is the Bayesian learning algorithm. The basic idea of probability distribution in Bayesian learning is to update the posterior probability of these random variables. Two main methods of Bayesian learning: One is to construct the Bayesian network manually by the application of summarized knowledge of domain expert, or from some received prior information. The other way to gain is by the means of data analysis. To construct actual structure of Bayesian network, we will make our best to keep the accuracy and rapidity learning by comprehensive complication of the experts’ knowledge and prior information, meanwhile combining with the method of data analysis and then update the prior probability to posterior probability.

RESULTS

To test the effectiveness of XCiot, we adopt the real-time data of XML of perceived equipment in the Internet of Things, and compare XCiot, XMill, XMLPPM, WinRAR, xwrt with each other based on the compressed test for the real-time data of XML. We compare the compression performance of XCiot by the analysis of compression ratio and time.

Experiment environment

CPU: Core i3-4000M 4, 2.40GHz and 4GB of RAM, the OS is Windows 8.1 Enterprise.

XML document

All XML documents of experiment are chosen from the perceived equipment in the Internet of

Things, the documents are nominated by the name of equipment, the unit is byte and shown in Table III.

TABLE III. XML DOCUMENTS

subway Power	Manipulator	Elevator	Packing Machine	Perceived station
4556	7670	9287	17409	37227

Most of the XML data of perceived equipment in the Internet of Things varies from several KB to dozen of KB. For example, there is a packaging machine, which is made of Siemens' PLC, its XML data probably only has a dozen of KB.

Compression ratio

The main purpose of file compression is to decrease the total volume of transmitting file. Compression ratio (CR): Defined as (1).

$$CR = 1 - \frac{sizeof(compressed\ \ file)}{sizeof(original\ \ file)} \quad (1)$$

After the compressed file divided the original file, we can get a ratio, the smaller ratio shows the higher compression ratio and greater compression effect. The test result of compression percentage among multiple compression algorithm for real-time data of XML is shown in Fig. 3.

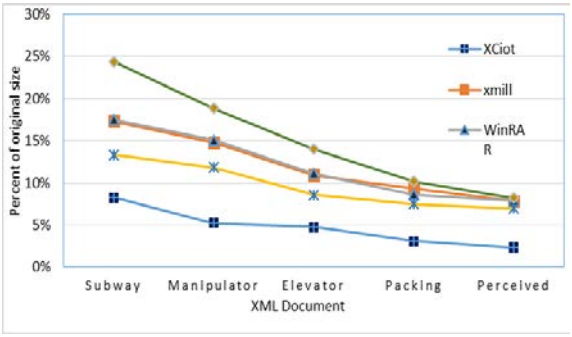

Figure 3. XML Real-time compression results

From the Fig. 3, comparing with others classical algorithm, we can find out XCiot has a better compression ratio for the real-time data of XML in the Internet of Things. It is due to the compression of time-related redundancy information by XCiot for real-time data of XML in the Internet of Things, thereby greatly reduce the size of compressed file.

Compression time

The data compression can cause the delay of data transmission in the process of compression, so in principle, the shorter compression time will be the better. The test result of compression time by the compression algorithm for real-time data of XML is shown in Fig. 4.

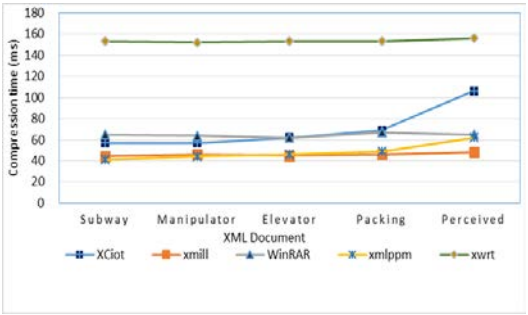

Figure 4. XML Real-time compression time

From the Fig. 4, we can find out the compression time of XCiot is shorter than xwrt compression method, but almost the same as other compression methods in the process of compressing smaller XML document. With the enlargement of XML document, the compression time of XCiot will be

longer than other compression methods, which is the result of the larger dictionary information by the dictionary compression of XCiot and longer time of inquiring dictionary in the process of compression. Make an index for dictionary can decrease the compression time of dictionary compression.

CONCLUSION

XCiot can make full use of time-related redundant information of XML file of perceived equipment in the Internet of Things, and greatly improve the compression performance through the study of pattern recognition algorithm to generate adaptive static dictionary. We need maintain the adaptive static dictionary regularly, so as to improve the self-adaptability of XML data compression in the Internet of Things, and achieve a better compression performance. As the compression side don't need to deliver the compressed dictionary to decompression side in the process of data transmission, which can improve the security of data to some extent. Although the compression time of XCiot is slightly longer than some classical compression algorithms, but they are in the same orders of magnitude and XCiot can greatly increase the compression ratio, which is of great value in compressing method for real-time data of XML of the Internet of Things.

Similarly, the XML data among each perceived equipment in the Internet of Things always contains some related redundancy information, and the data fusion process of perceived equipment may also further increase the compression. In the further work, we plan to focus on the data fusion and compression among the perceived equipment in the Internet of Things, and thereby can improve the transfer efficiency of XML data in the Internet of Things.

Acknowledgment

This work was financially supported by the National Natural Science Foundation (61201246), and National Natural Science Foundation through the General Programs (51178368, 51478372). We are also very grateful to Yang Meng, Jianxin Zhou and Qunwu Lv had provided helpful suggestions.

References

[1] Li Guan, "Research on the data fusion of indoor environmental monitoring based on Bayesian network," Changchun: Jilin University, 2013.

[2] Sherif Sakr, "XML compression techniques: A survey and comparison", Journal of Computer and System Sciences Volume 75, Issue 5, August 2009, Pages 303–322.

[3] David Salomon, "Data Compression: The Complete Reference," pub-SV, 2004.

[4] Shanshan Zhang, "Research on the technology of XML compression," Wuhan: Huazhong University of Science and Technology, 2011.

[5] Liang Li, "Research on the compression of routing algorithm based on Contextual data in Internet of things," Dalian: Dalian University of Technology, 2013.3.

[6] Moore, J.P.T.; Kheirkhahzadeh, A.D.; Bagale, J.N., "Domain-Specific XML Compression," Data Compression Conference (DCC), 2013 , pp.510,510, 20-22 March 2013.

[7] Dongping Wei; Xiangli Wei, "Structural Join Oriented XML Data Compression," Software Engineering (WCSE), 2012 Third World Congress on , pp.29,33, 6-8 Nov. 2012.

[8] Mansour, A.M.A.; Fouad, M.A.M., "Dictionary based optimization for adaptive compression techniques," Information & Communication Technology Electronics & Microelectronics (MIPRO), 2013 36th International Convention on, pp.421, 425, 20-24 May 2013.

[9] Xiong, Tao; Suo, Yuanming; Zhang, Jie; Liu, Siwei; Etienne-Cummings, Ralph; Chin, Sang; Tran, Trac D., "A dictionary learning algorithm for multi-channel neural recordings," Biomedical Circuits and Systems Conference (BioCAS), 2014 IEEE , pp.9,12, 22-24 Oct. 2014.

[10] Jianjia Wu and Wei Zhao, "From Net of Things to Internet of Things," Computer research and development, 2013, 06:1127-1134.