Chapter 12
Conclusions and Future Work

Badya Al-Hamadani
University of Huddersfield, UK

Joan Lu
University of Huddersfield, UK

ABSTRACT
This thesis introduces a new model which has the ability to compress an XML document efficiently and retrieve information from the compressed file according to vague queries and even various other types of queries. This chapter will outline the main conclusions of the research as well as the main advantages and limitations of the designed model. Finally, the chapter will also list possible future trends in this research in terms of developing the proposed model.

1.1 CONCLUSION
As the importance of XML usage for storing and transferring data via the World Wide Web becomes increasingly clear, there is a corresponding need to compress the size of XML documents, dealing with them in their compressed mode so as to make them accessible to devices with limited resources. When these compressed documents are used by simple users, in a situation where there is absence of schema, or if such a user has no exact idea of what s/he is looking for, there should be a special technique available to adequately deal with these types of queries. The questions had been raised by this research and their answers are as follows:

1. Is it possible to design a new compression technique that has the ability to compress the XML documents and achieve better compression ratio without the need to the document’s Schema or its DTD?

The answer to this question is XCVQ compressor. The design of the model showed the best average compression ratio (78.45) among the other XML queriable compressors without the need to the XML schema to be available. This was due to several reasons, such as: (1) limiting the storage of each element and attribute name in the document

DOI: 10.4018/978-1-4666-1975-3.ch012
Conclusions and Future Work

to only one number, which represents the order of that element or attribute in the XML document, instead of being two numbers, and (2) increase the granularity of the data to be compressed in order to perform better compression ratio. Although this design issue increased the compression ratio, but it affects the time required to compress the document by increasing this time to be higher than the time required to compress these documents using other techniques. However, the compression process usually made only once, while the querying process can be done hundreds of times to retrieve information from the compressed files.

2. What is the influence of the structure redundancy on the overall size of the XML document?

To answer this question, XCVQ Structure Compressor was designed. In the compression process of the XML documents, the research found the strong affect of the redundancy in the structure of the document on its overall size. By succinctly storing the structure part of the XML document and keeping the data part as it is, the experiments showed good compression ratios which were up to 85.43 and averaged 49.47 for the tested XML corpus. This shows the big redundancy in the structure part of the document, apart which is considered to be very important for several purposes and retrieving information is one of them.

3. What are the main types of vague queries and when they can be occur? Have the existing XPath query language the ability to answer vague queries? If no, what is the required expansion that should be made on XPath to give it this ability?

Vague queries are one of the important types of queries. They occur in different situations and require special ways to be processed since the existing query languages do not have the ability to answer these queries. The XCVQ-QP can deal with simple and complex queries by forcing each query to pass by two decomposition stages in order to make it easier to retrieve information from the relevant document(s) and then combine the sub-results to be decompressed and submitted to the user. This process required the expansion of XPath query language in different sides: the path expansion, the data value expansion, and the set of functions expansion. The time required to process the queries are very competitive especially when dealing with structure-based queries, since the compressed structure of the document helps in accelerating the retrieving process.

4. How to determine the relevant XML document(s) from thousands of documents without the need to scan them completely for time saving purposes? And is it possible to retrieve information from more than one XML document without the pre-specification of these documents using one XPath query?

Instead of scanning the complete document to search for a specific bit of data, XCVQ-QP uses the path-dictionary, which contains all the elements and attributes names, to specify the relevant documents from thousand of XML documents. In this way, it is now possible to retrieve information from unspecified document(s). While all the existing XML query processors required the user to pre-specify the required documents to retrieve information from them, XCVQ-QP has the ability to retrieve information from one or more than one XML document without the need to specify exactly which document could contain the required information.

1.2 RECOMMENDATIONS

• The main purpose of designing XCVQ is to process vague queries on compressed XML documents. For that reason, the first