Advances in Digital Technologies
J. Mizera-Pietraszko et al. (Eds.)
IOS Press, 2017

doi:10.3233/978-1-61499-773-3-51

Evaluation and Comparison of Metrics for XML Schema Languages

Oluwadamilare FALOLA[a,1] Sanjay MISRA[a] Adewole ADEWUMI[a] and Robertast DAMASEVIČIUS[b]
[a] Center of ICT/ICE Research, CUCRID Building, *Covenant University, Ota, Nigeria*
[b] *Kaunas University of Technology, Kaunas, Lithuania*

Abstract. The importance of XML (eXtensible Markup Language) can't be understated; their usefulness may range from data sharing to data transport in software systems. Schema languages describe the structure of an XML document and the common schemas languages are Document Definition Type (DTD), W3C XML Schema and RelaxNG. Applications depend heavily on XML documents to be free of error and this makes it imperative to determine the quality of such schema document. Schema metrics is used to achieve this, and several of them have been proposed in recent years. In this paper we present the existing schema metrics and to make comparative studies on all schema metrics, figuring out the features, advantages and limitations of each metrics.

Keywords. XML schema, XML metrics, Document Type Definition, XML schema

1. Introduction

XML schemas describe the structure of an XML document and are used for validating XML documents. They define a bond that provides a base for an application to use the XML data [1]. XML schema could also be referred to as a grammar that specifies a language that constrains and documents corresponding XML. There are several types of schema languages, which are XML DTD, XML Schema, XDR, SOX, Schematron, and DSD. XML schema, DTD and RelaxNG are more common.

The XML document is probably the smallest part of the software but its influence on the software project as a whole can be quite profound [2]. Therefore the quality of these documents must be ensured. Given an application that uses XML as its internal data representation, if its structure is poorly formed, it will definitely affect the results of the application. IEEE defined quality "as the degree to which software possesses a desired combination of elements". These elements as given by ISO 9126 include: functionality, reliability, efficiency, usability, maintainability, and portability [3]. Software metrics is the tool software developers' use to measure software quality. It can be defined as the measurement of aspects of construction and testing that result in an ability to describe the quality, cost and value of the software. XML schema metrics

[1] Corresponding Author, Sanjay Misra, Department of Electrical and Information Engineering, Covenant University, Ota, Ogun State, Nigeria; E-mail: sanjay.misra@covenantuniversity.edu.ng.

are designed to enable the quantification of the schema size, complexity and quality of the XML documents [4].

There are really no genuine metrics developed specifically for XML, most metrics used today for XML schemas were adopted from software engineering world [6]. Examples of metrics includes the size metrics i.e. metrics that count individual elements of a schema document, structural metrics, i.e. metrics that use graph data structures to try and decipher the structure of a given schema document. These metrics are quite simple. The more sophisticated metrics try to evaluate by looking at the internal structure, recursion, and polymorphisms.

A metric is not certified for use until it is validated, as is the case in software engineering. XML schema metrics must thus be validated theoretically and empirically. The rest of this paper is structured as follows: Section 2 discusses the existing schema metrics. In section 3, we describe the methodology adopted in this study while section 4 presents the results obtained with discussion. Section 5 concludes the paper.

2. Existing Schema Metrics

In this section we describe the existing schema metrics e.g. DTD, XML schema and Relax NG.

2.1. DTD Schema Metrics

Schema metrics that fall under this category include those proposed by Klettke et al. [5] and Misra et al. [7].

From literature [4], the first step in XML metrics research began with Klette et al. In this paper several sets of metrics were proposed for DTD documents in order to measure complexity of XML documents. The metrics include: size, structural complexity, structure depth, fan-in and fan-out measures.

Misra et al. [7] went on to propose entropy metric and distinct structured element repetition scale metric for measuring the complexity of DTD schema languages.

For the entropy metric, chunks were used to represent a group of related items. In schema documents chunks will be the declared elements and parent-child relationship will be server as relationship among other elements. Also attributes can be referred to as chunks since they are part of the element declaration. In this metric, the graph data structure is used to represent the schema document, where nodes are represented as elements and attributes the edges that connect them are the parent-child relationship. In this case the graph is a directed graph. In relation to the definition of chunks above, chunks are represented by a directed graph. For each element in the graph we find their fan-in, fan-out and number of attributes and with these values, the elements are grouped into equivalent classes. Any element that has the same value in any of these three are said to have the same structural complexity.
The entropy for the document type definition is:

$$E(DTD) = -\sum_{i=1}^{n} P(c_i) log_2\ P(c_i) \quad (1)$$

Where n is the number of equivalent classes.
P (Ci) denotes the probability of class Ci

The Distinct Structured Repetition Scale (DSERS) metric is given as:

$$DSERS\ (DTD) = \sum_{i=1}^{p} \frac{dei^2}{\#e} \quad (2)$$

Where p is the number of equivalent classes.
dei is the number of members inside the ith class
#e is the total number of element nodes in the graph representation of DTD.

2.2. XML Schema Definition Metrics

The XML Schema Definition (XSD) metrics found in literature include: XML schema complexity and quality [2][4], Lammel's size metrics [8][10], structure metrics for XML schema.

An XML complexity and quality index for XSD documents was built in [4]. The work done in [2] was based on using elements of the XSD documents to determine the quality of the XML Schema document.

In [4], proposed metrics that are used to measure the structural properties of XML schema documents. In his work, he used graph representations to shows dependencies and relationships between components of an XML schema document. The metrics dependency graphs; such as successor graph (SG), directed acyclic graph (DAG), strongly connected components graph (CG).

The metrics proposed by Lammel et al. [8] include: file size in kilobytes, XSD-agnostic schema size, XSD-aware counts, McCabe cyclometric complexity, depth and breadth of content models, code-oriented and instance-oriented breadth/depth.

In [9], a metric is proposed based on the complexity value of the XSD document. The work originates from [6]. It was evaluated by taking into consideration all the complexity values of each component. Every component has a weight value assigned to it via its internal architecture and with this value the degree of the component complexity is evaluated.

In [10], it was pointed out that, given the size and complexity metrics, one should be able to categorize the size and complexity of a given schema, compare different schemas with regard to size or complexity and compare different versions of the same schema.

2.3. Measuring Qualities of XML Schema Documents

The study in [10] proposed three metrics for measuring the quality of XML schema documents. They include Reusable Quality metric (RQ), Extensible Quality metric (EQ) and Understandable Quality metric (UQ). They were proposed to measure the Reusable i.e. defining the components globally and can be leveraged by other XML schemas, Extensible i.e. ability for developers to write extension schema by adding additional features to the original in a controlled way and Understandable (clear, consistent and unambiguous schemas) of XML schema document in web engineering process respectively. These metrics were formulated based on the binary entropy function and rank order centroid method.

2.4. Complexity Metric for XML Documents

The study in [11] proposed a new metric for finding the complexity of XSD schemas due to the weakness of other metrics proposed before e.g. size metrics, index metrics. This metric extended their formal metric by adding the ability to evaluate the complexity of an XSD schema document when there are recursive definitions in it. They also stated that the complexity of a XSD document is dependent on the complexity degree, which a weight is assigned to each internal component to reflect its complexity. The sum of all the internal weights provides the complexity value of the XML schema.

Metrics definition:

In [11], five factors were listed that affects the complexity of an XML schema document. Total complexity of the XSD document:

$$C(XSD) = C(V_g) + C(G_g) + CT_g \tag{3}$$

C (Vg): total complexity values of all unreferenced global elements and attributes.

$$C(V_g) = C(E_g) + C(A_g) \tag{4}$$

Where C (Eg) and C(Ag) denote the complexities of global elements and attributes definitions.

C(Gg): total complexity values of unreferenced global elements and attributes group.

$$C(G_g) = C(EG_g) + C(AG_g) \tag{5}$$

Where C(EGg) and C(AGg) denote complexities of global elements and attributes group definition.

C (Tg): total complexity values of unreferenced global complex and simple type definitions/declarations.

$$C(T_g) = C(cTg) + C(sTg) \tag{6}$$

Where C(cTg) and C(sTg) denotes complexities of global complex and simple type definition.

2.5. Complexity Metrics OGC Web Services Case Study

The OGC- web service uses XML schema to pass information among its users. This huge dependence on XML schema makes it imperative to ensure the quality of such documents. In [12] work, they used the existing complexity metrics to measure the quality of their XML schemas; they also proposed three metrics that computes the

complexity of a XML schema by checking the influence of subtyping in the XML documents. They include:

2.5.1. Data Polymorphism Rate: computed with the formula

$$DPR = \frac{\sum_{I=1}^{N} PE_{CT_i}}{\sum_{j=1}^{N} ECT_j} \quad (7)$$

DPR shows the measure of polymorphism in a schema document.
Where
N is total number of complex types.
PECTi PE_(CT_i): number of elements in the declaration of complex type that are polymorphic.

2.5.2. Data Polymorphism Factor (DPF)

This calculates the influence of polymorphism on the schema complexity.

$$DPF = \frac{\sum_{i=1}^{N} OE_{CT_i}}{\sum_{j=1}^{N} E_{CT_j}} \quad (8)$$

OECTi: number of possible different elements that could be contained in a complex type.

2.5.3. Schemas Reachability Rate

This metric aims at finding the proportions of imported schemas that are not explicitly referenced [13].

$$SRR = \frac{|V_{Rm}(G_{SH})| - |V_{Rm}(G_S)|}{|V_S|} \quad (9)$$

GS: directed graph (Vs, Es), Vs includes all global elements declared.
GSH: directed graph (VsH, EsH); VsH, EsH both extend Vs, Es.
VRm (G): directed graph (V, E) and Vm.

3. Methodology

Based on the various XML schema metrics for DTD and W3C XML schemas identified in the previous section, this study discusses them based on their unique features, advantages and limitations. In addition, this study also discusses whether or not theoretical, practical and empirical validations have been conducted on the various metrics.

Theoretical validation is mainly used to check that a given metric actually does as it claims. It measures the ability of the metric to achieve what it purports to do. Representation condition, scale measurement, extensive structure and Weyuker's properties are some of the suggested ways of performing theoretical validation [12][16]. Some other approaches also exist such as self-organizing map [10]. On the other hand, practical validation is performed using Caner's framework.

Empirical Validation involves checking how useful a metric is in relationship to its domain. This can be done using small examples, case study, or big projects from the Web or by considering real projects from the industry.

4. Results & Discussion

In this section, the various complexity measures are grouped based on authorship so as to facilitate discussion on their unique features, advantages, limitations and the type of validation that has been conducted on them.

4.1. Klettke et al. complexity measures

They employ graph data structure in their description. In particular, size and structure complexity are applied to the whole DTD schema, while the rest are applied to each element and attribute. They can help to detect if a schema document is understandable and usable by helping to identify which elements are difficult to understand.

The drawback of these measures is that they can only be used on DTD schema documents. Also, the metrics are simple and general metrics thus making them unsuited for application in complex documents. Furthermore, the size metric has been found to be inaccurate. In addition, all the five measures are dependent on each other.

As regards validation, no forms of theoretical or practical validations have been conducted on the measures. Empirical validation has been carried out using two small examples.

4.2. McDowell et al. complexity measures

These measures were proposed to exploit the advanced features of XML schemas. They were derived from existing software engineering metrics and are easy to understand and quantify. As a result, tools have been developed for the measures.

Although, two indexes were proposed for interpreting the results of the measures, these indexes can be very simple and inaccurate. Furthermore, the quality indexes defined do not capture the full quality model defined in the ISO quality model. In addition, the results obtained are similar to results of other metrics.

As regards validation, no forms of theoretical or practical validations have been conducted on the measures just as with the Klettke et al. measures. Empirical validation has however been conducted using one example.

4.3. Visser Joost complexity measures

These measures were developed to measure structural properties and are based on graph data structures. A number of the measures were derived from software artifacts and as a result a prototype tool has been developed for the schema metrics.

Similar to the first two measures, no forms of theoretical or practical validations have being conducted save for an empirical validation that was conducted using one small example. As a result, the measures cannot be adopted in the industry due to lack of validation.

4.4. Thaw et al. complexity measures

These metrics were aimed at measuring the reusable, extensible and understandable qualities of XML schema documents. The measures were formulated based on binary entropy function and rank order centroid method.

The drawback of these measures is that no software tool has been implemented to aid in their measurement thus making it complex and unlikely to be adopted in the industry or by other researchers. In addition, it has only been tested with XML schema.

As regards validation, unlike the previous measures, this measure has been validated theoretically through Weyukers' properties for which it satisfies eight conditions as well as through self-organizing maps. Though it has not been validated practically, empirical validations have been conducted using 100 small examples, as well through the use of big projects from the Web.

4.5. Misra et al. complexity measure

This measure aims at measuring complexity of an XML schema through its internal structure and recursion. It was derived from the authors' previous work and so the measure is dependent on the complexity values of each of the internal components.

The drawback of this measure just like the Thaw et al. complexity measures is that no tool has been developed to automate the measurement process. Also the measure has not been applied to DTD schema even though DTD schema is not.

As regards validation, similar to the Thaw et al. model, it has been validated theoretically using Weyukers' properties and satisfied six of the properties. In addition empirical validation has been done using 50 small examples.

4.6. Tamayo et al. complexity measures

These measures were developed for measuring the influence of subtyping on complexity of XML schemas used in web services messaging. It uses the graph data structure.

The limitation of these measures include that no software tool has been developed to facilitate the measurement process. Also, the measures have not been tested with RELAXNG.

As regards validation, only empirical validation via case study has been performed using the measures.

4.7. Misra et al. complexity measures [12][17]

These measures were targeted at finding the structural complexity of DTD schema languages. The schema document entropy metric E(DTD) was adopted from communication information theory as described by Shannon [14]. Distinct structured element repetition scale DSERS (DTD) was adopted from ARS metric, as described by [15]. However, these measures have only been applied in the DTD schema language.

As regards validation, it has been validated theoretically using ratio scale measurement and empirically using small examples (65 small examples) and also applied in big projects from the web.

5. Conclusion

XML are of great value in software engineering systems, their uses range from data interchange to data structuring. XML schema defines the structure of an XML document. Deformed XML schema, can in the end affects the functionality and usability of a software system. High attention has to be paid to the metrics used for validating XML documents. In section 4 we outlined the works of the seven authors and metrics reviewed and listed how well these metrics were validated. Among these metrics only few performed an elaborate validation of their metrics and they are mainly the recent ones. Metrics that have been proposed earlier have no information about metrics validation.

Also amongst those that validated their metrics, none did all three validations. The study in [10] used Weyuker properties with their metrics passing eight out of eleven metrics for their theoretical validation. They also used self-organizing maps, a machine learning method to also validate their metrics. Empirical Validation is considered to be imperative for all new metrics because it tests applicable a metric is to real life examples.

In our review, all the authors validated their metrics empirically. Some of them were shallow validations as their metrics were only tested with one XML schema. A good example of this is [17] in which they used approximately 65 XML schemas downloaded from the web and testing their metrics with it. Another example is found in [12]. Here the authors used OGC web service as a case study when validating their metrics. They noticed the size of their schemas documents were growing very large and it became complex for developers to use, so they developed metrics to measure the complexities and therefore ensured the quality of their schemas.

The quality of an XML schema document must be ensured if it is to be utilized in the industry. The developer of software when using XML metrics can easily detect malformed or incorrect schemas. Overall, XML schema metrics helps in the end to improve the quality of a software system. In this paper we have given a comparative study of the schema metrics available today. We have also presented several metrics, and the metrics that have been validated.

Acknowledgements

We acknowledge the support and sponsorship provided by Covenant University through the Centre for Research, Innovation and Discovery (CUCRID).

References

[1] C. Binstock, D. Peterson, M. Smith, M. Wooding, C. Dix, C. Galtenberg, The XML Schema Complete Reference, Addison Wesley, 2002.
[2] A. McDowell, C. Schmidt, & K. Yue, Analysis and Metrics of XML Schema, Software Engineering Research and Practice (2004), 538-544.
[3] R. Pressman, Quality concepts. Software Engineering: A Practitioners' Approach (7th ed.), McGraw-Hill, New York, 2010.
[4] J. Visser, Structure metrics for XML Schema. Proceedings of XATA, 2006.
[5] M. Klettke, L. Schneider, & A. Heuer, Metrics for XML document collections, In International Conference on Extending Database Technology, Springer, Berlin, 2002.
[6] B. Sumak, M. Hericko, & M. Pusnik, Towards a framework for quality XML schema evaluation, In 29th International Conference on Information Technology Interfaces, (2007) 783-788.
[7] D. Basci, & S. Misra, Document Type Definition (DTD) Metrics, Romanian Journal of Information Science and Technology, 14 (2011), 31-50.
[8] R. Lämmel, S. Kitsis, & D. Remy, Analysis of XML schema usage, In Conference Proceedings XML, 5 (2005).
[9] D. Basci, & S. Misra, Entropy as a measure of quality of XML schema document, The International Arab Journal of Information Technology, 8 (2011), 75-83.
[10] T.Z. Thaw, & M.M. Khin, Measuring qualities of XML Schema documents, Journal of Software Engineering and Applications, 6 (2013), 458.
[11] D. Basci, & S. Misra, Measuring and evaluating a design complexity metric for XML schema documents, Journal of Information Science and Engineering, 25 (2009), 1405-1425.
[12] S. Misra, Cognitive complexity measures: An analysis, IGI Global Publishing, USA, 2011.
[13] A. Tamayo, C. Granell, & J. Huerta, Analysing complexity of XML schemas in geospatial web services, In Proceedings of the 2nd International Conference on Computing for Geospatial Research & Applications (2011), 17.
[14] C.E. Shannon, A mathematical theory of communication, Mobile Computing and Communications Review, 5 (2001), 3-55.
[15] M.A. Boxall, & S. Araban, Interface metrics for reusability analysis of components, In Proceedings of Australian Software Engineering Conference, (2004) 40-51.
[16] L. Briand, K. El Emam, & S. Morasca, Theoretical and empirical validation of software product measures, International Software Engineering Research Network, Technical Report ISERN-95-03 (1995).
[17] D. Basci, & S. Misra, Complexity metric for XML schema documents, In Proceeding of 5th international Workshop on SOA, 2007.