Available online at www.qu.edu.iq/journalcm
JOURNAL OF AL-QADISIYAH FOR COMPUTER SCIENCE AND MATHEMATICS
ISSN:2521-3504(online) ISSN:2074-0204(print)

# Compression-based Aggregation and Clustering Techniques for XML Enhanced Performance

***Suad kamil Ayfan[a]*, Ahmed M. Mahdi[b]***
[a,b] *College of Computer Science and information technology, University of Al-Qadisiyah, Iraq*
*.Email: it.mast.23.11@qu.edu.iq, ahmed.m.mahdi@qu.edu.iq*

ARTICLE INFO

*Article history:*
Received: 19 /05/2024
Rrevised form: 23 /06/2024
Accepted : 27 /06/2024
Available online: 30 /06/2024

*Keywords:*

web services , aggregation, soap message, XML, clustering, compression

ABSTRACT

This paper has presented a comprehensive literature study and discusses the primary problem of network congestion and bottlenecks in XML applications. The explosive growth of Web services has caused a significant problem in network traffic due to the largely repetitive XML textual structures. The challenge of XML message size reduction in web services has been examined, focusing on new approaches such as compression methods and message clustering for Aggregation and compression of XML messages efficiently. The goal is to examine how different strategies affect network traffic. This overview discusses twenty-five publications categorized into four: compression techniques, aggregation approaches, clustering for web services aggregation approaches, and other web services clustering models. In comparison to all models, evaluation has shown that web service clustering solutions efficiently reduce traffic. Technically, the web service clustering for aggregation models has achieved an excellent average compression ratio of up to (14.12).

MSC..

https://doi.org/10.29304/jqcsm.2024.16.21546

## 1. Introduction

Web services, a pivotal middleware software category, provide essential Internet access to networked resources through fundamental network protocols and mechanisms such as HTTP and TCP[1][2]. These services enable communication and data exchange between disparate systems over the Internet, making them indispensable in modern IT infrastructures. The widespread adoption of Cloud Web services is facilitated by the inherent interoperability and self-descriptive nature of XML-based interfaces [3][4], which allow seamless integration across various platforms and applications[5]]. The Extensible Markup Language (XML) forms the backbone of SOAP (Simple Object Access Protocol), a protocol used for exchanging structured information in web services. HTTP and TCP are the primary protocols to transmit these XML messages, ensuring reliable and efficient communication[6][7].

The Web Services Description Language (WSDL) is crucial in defining a web service's public interface, essential for achieving interoperability and facilitating integration in distributed systems. WSDL provides a standard way to describe the services offered by a web service, including the format of the messages, the transport protocols, and the

∗Corresponding author *Suad kamil Ayfan*

Email addresses: *it.mast.23.11@qu.edu.iq*

Communicated by 'sub etitor'

endpoint addresses[8]. Moreover, the Universal Description, Discovery, and Integration (UDDI) standard is employed to publish and discover web services, further enhancing their accessibility and usability[9] [8]]. Despite the numerous advantages offered by web services, they encounter significant challenges, particularly related to the substantial payload of XML messages. XML, while highly flexible and extensible, is verbose and can lead to considerable network congestion and bottlenecks. The repetitive nature of XML tags and attributes increases the size of the messages, causing higher bandwidth consumption and longer transmission times[10]. This verbosity can severely impact overall performance and scalability, especially in environments with limited network resources or high traffic loads.To address these challenges, various techniques have been proposed to reduce the size of XML messages and enhance network performance. These techniques include compression methods, which aim to decrease the redundancy and size of the XML data, and message clustering for aggregation, which groups similar messages together to improve efficiency. By effectively reducing the size of XML messages, it is possible to alleviate network congestion, improve response times, and enhance the overall performance of web services[11][12]. Generally, This survey makes significant contributions to the field of XML message size reduction in web services. It offers a comprehensive classification of existing studies into four main categories: compression techniques, aggregation approaches, clustering for web services aggregation, and other web services clustering models. The survey provides a critical analysis of each category, highlighting the strengths and weaknesses of different approaches. Additionally, it identifies several gaps in the current literature, suggesting areas that require further investigation to improve network efficiency in XML applications.

### *1.1. Motivation*

The network performance has been affected due to XML-based web message representation, principally as a result of the significant network congestion caused by their considerable redundancy[12]. Consequently, Web services based-soap suffer bottlenecks and congestion as a result of Web messages being larger than the real payload [13]. In sum, many times, network response time is very low, delaying user's services [14]].

### *1.2. Existing Solution*

This paper provides an inclusive overview of the web services domain, a subject that has received many studies and research interests. Among the challenges noted in this domain is the substantial payload of web messages. Because of this issue, major latency and congestion concerns are now inside the network architecture. Consequently, there has been a significant need to develop various solutions to address these network-related concerns. Technically, in this paper, twenty-five distinct approaches are described and categorized into four groups based on the applied techniques. In practical terms, these approaches fall under the following classifications: Web-based service clustered is used for aggregating, as well as for other purposes. In addition, aggregating techniques and compressed techniques.

### *1.3. . Evaluations techniques*

The Compression Ratio (CR) metric is the primary measure used in this survey report to assess the quality of previous research. As a result, CR values are gathered from all the included articles and arranged into tables that match the selected analysis categories. Compression ratio (CR) is a ratio that measures the extent to which the size of the original data is reduced after applying compression techniques. It is calculated by dividing the size of the original data by the size of the compressed data [15]–[17]. Clustering for Aggregation, Aggregation, in addition to compression approaches, are compared within their specific categories as an essential concept. Additionally, summary statistics, such as the average CR, are provided for each approach group and contrasted with other groups to analyze the models' performance. Technically, clustering of the fractal similarity approach has consistently outperformed other techniques for improving traffic networks in terms of CR, achieving up to (21.70) with v. large messages. Furthermore, studies on Two-bit and One-bit aggregating models have demonstrated their exceptional performance in terms (of CR) when compared to alternative aggregation methods. These models have achieved a maximum performance of 20.26 with significantly high message sizes.

### *1.4. Paper Organization*

This paper is organized and coordinated as Section 2 clarifies Compression approaches. Section 3 describes aggregation approaches for web services. Then, clustering web services for aggregation methods are presented in section 4. Section 5 illustrates other WS clustering models. Analysis and evaluation are presented in section 6. Finally, section 7 describes the conclusion.

## 2. compression approaches

Before a message is sent across the network, its size can be decreased using compression techniques. These choices may significantly impact response time and throughput.

The issue of huge XML message sizes resulting from the presence of repeated tags inside the XML message structure has been discussed by GYAN.P et al. [14]. A novel compression has been proposed. Both XML and SMCA XML compression techniques have been investigated with the aim of determining their weakness. The proposed model is based on rebuilding the XML messages in a numerical form. Compression ratio and time were used for evaluation.

Moreover, SMCA and other techniques are used to benchmark the proposed model. Two datasets have been used for experiments with various message sizes. The obtained best results are related to the original message size as it has achieved about 10, 16, 21, and 22 CR for small, medium, large, and very large messages, respectively. The proposed model suffers from the possibility of receiving a large number of XML characters that may suspect the model. Furthermore, the results obtained show lower efficiency on large messages than other techniques.

Another approach achieved by Hejun Xu et al. [18]aimed to apply a compression technique capable of reducing redundancy and inefficiency in Bim information, especially in the ifcxml data format used for data exchange between project participants and through different software. In this paper, an ( iterative reference mapping method (IRMC)) is used To compress the IFCXML format depending on the classification of duplicated information. That cleans up the structured file and enables a lightweight Bim model. To test the compressor's performance using two metrics: compression ratio (CR) and Time compression ratio (TCR). The datasets are used to evaluate the proposed method with six sample BIM models. That was sourced from training software for the Chinese BIM Skill Level Examination, sample files from a mainland Chinese architectural design institute, and the Open IFC dataset. The cr of IRMC for six samples was 18 %-59 %. The study only concentrates on the compression of IFCXML files and does not consider other file formats used in the BIM domain.

Athanasios Kiourtis et al. [19] have proposed a D2D protocol to address the problem of health information exchange between smartphone devices and health information systems. Incredibly, the most exciting HIE techniques are proprietary and non-interoperable across vendors, causing difficulty for citizens and healthcare practitioners in exchanging healthcare information. Furthermore, accessing healthcare data often needs an internet connection. That research paper has suggested a new protocol, D2D (device to device), based on Bluetooth. Technically, it has been designed to deal with open specification and lossless compression. Overall size, duration, and Total time for interaction and transmission are the metrics used to measure the effectiveness and performance of the D2D protocol when sharing various datasets of diverse sizes. Empirically, eight different datasets with various sizes and types have been used to evaluate the proposed device-to-device protocol for exchanging information. The datasets have included textual, image, audio, and multimedia data. Clearly, The Compression technique compacts the exchange data more than ( 60% )of their first size without losing data; the result minimizes the all-time by 75%, evidently, in a vast dataset. In fact, the proposed D2D protocol was evaluated in the real world using sample data, and it is unclear how the protocol would perform in a more scale implementation.

Another study by Danny Hucke et al. [20]] discussed the problem of Grammar-based tree compressors facing challenges when dealing with tree structures with high complexity and low redundancy—for example, compression XML data. Bounded entropy can enhance the effectiveness by providing a more predictable environment for identifying and representing patterns within the tree data. Applied to extend entropy bounds for grammar-based compression from strings to trees. The authors propose a new notion of kth-order empirical entropy for node-labeled binary trees that captures regularities in both labels and structure. They show that a binary encoding of tree straight-line programs (TSLPs) yields binary tree encodings of size bounded by the kth-order empirical entropy. The compression ratio has been adapted to test that method. All datasets are used and are available from http://xmlcompbench.sourceforge.net/Dataset.html. The paper focuses on grammar-based tree compression for node-labeled binary trees. However, it does not address the compression of other types of trees or tree structures.

The problem of the high network traffic and congestion caused by huge XML messages in SOAP Web services Has been addressed by Shammarya et al. [[21]]. This research has utilized two techniques. The first technique is fixed-length encoding. The authors have built an XML tree from the message and converted it into a binary tree. As a result of removing closing tags, a high compression ratio has been achieved. The second technique is Huffman encoding, which is a variable-length encoding method. The authors also construct an XML tree and binary tree. However, this particular technique offers the highest level of efficiency in the compression of large data. The efficacy of the proposed

approach has been evaluated by using the average compression ratio (Av. CR). A total of 160 SOAP messages have been utilized for the purposes of testing and experimenting. The messages exhibit a wide range of sizes, spanning from 140 bytes to 53 kilobytes. The optimal results for AV and CR are, in order, (2.037, 3.015, 7.8, and 13.45). The scope of the experimental results related to the testing and comparing the proposed methodologies is constrained to message sizes between 140 bytes and 53 Kbytes. This limited range may not encompass the message sizes observed in real-world circumstances.

Chengxi Bernad Siew et al [[22]Have discussed the disadvantages of City Geography Markup Language (CityGML) used to exchange Spatial Data Infrastructures due to its XML-based format, which leads to reduced data transfer efficiency and storage issues. The use of CityGML is constrained by its inefficiency in terms of file size and bandwidth consumption. The paper has provided a solution called CitySAC (CityGML Schema Aware Compressor), a compression system designed for efficient data transactions within Spatial Data Infrastructures (SDIs). The system encodes the CityGML data using a dictionary approach, maintaining queryable advantages and providing partial decompression capability. It also allows partial decompression of the document for a specific area of interest. The proposed method is evaluated by computing the compression ratio and time compression. Technically, the CitySAC system on 6 CityGML datasets (Commercial Building, National Audit, Putrajaya Convention, Putrajaya Mosque Seri Gemilang, and Putrajaya All) have been investigated. The encoding procedure resulted in compression rates exceeding 90% for all examined datasets, leading to a reduction in size to around 7-10% of the data's size.

In comparison with (fast, LZMA, and BZIP2) techniques. The integrated model with LZMA has provided better temporal results than the rest. Limited applicability: The CitySAC system is specifically designed for the CityGML format. It may not be suitable for other types of spatial data or data formats—dependency on LZMA is only needed to achieve better compression results. , the system's performance is based on the performance and availability of LZMA.

Another study by Hariharan Devarajan et al [23]Have discussed the increase in data like XML data, in addition to other types within modern applications, puts significant pressure on storage systems. To solve this, developers employ data compression techniques to reduce size to limit that problem. Moreover, The problem of choosing compression libraries depends on the kind and format of the input data, making it difficult to select the best compression library for specific work. That paper has Presented ( Ares) a flexible, intelligent, and adaptive compression framework that can dynamically select a compression library for a given input set based on the nature of the workload and offers users the necessary infrastructure to Set the selected library. Furthermore, that framework enables the user to add more compression libraries. In order to evaluate the utilized framework, compression time (CT), decompression time (DT), and compression ratio (CR) are measured. The research paper tests the effectiveness of several compression libraries using pre-generated datasets. Characters, integers (with short, long, signed, and unsigned modifiers), sorted integers, floating point numbers, and double floating point numbers are all included in these datasets. In addition, several data formats, such as columnar data (e.g., Avro, Parquet), scientific data (e.g., HDF5), textual data (e.g., CSV, JSON, XML), and binary data (e.g., POSIX). Ares proved the ability to dynamically choose the best compression library based on the workload type and the input data's characteristics. It has achieved a balance between compression time (CT), decompression time (DT), and compression ratio (CR). All operations are performed faster than other compression libraries. It has provided a performance boost of 6.5x over bsc, 4.6x over bzip2, and 5-40x over lz4, quickly and snappy while maintaining a CPU utilization of 58% and a memory utilization of 64%. The model has yet to be proven effective on big data or high-throughput workloads.

## 3. Aggregation approaches

web message aggregation has emerged as a prominent method for resolving network bottlenecks and congestion. Effectively decrease network volume by aggregating messages and deleting unnecessary information.

The problem concerning the performance of XML message representation in web services has been addressed by Dhiah Al-Shammary et al [1]]. Technically, it might be argued that these messages result in bottlenecks and congestion in network traffic due to their size exceeding the payload of the requested services. To tackle this issue, the researchers present two novel SOAP Web message aggregation models, namely the Two-bit and One-bit aggregation techniques. These models aim to aggregate several messages into a single message by employing a two-step process: grouping similar messages based on their similarity and subsequently applying the aggregation technique. Two evaluation measures are applied: compression ratio (CR) and processing time. For the purpose of evaluation, a dataset consisting of 160 authentic SOAP Web messages has been utilized. These messages have been categorized into four distinct classes based on their size, namely small, medium, large, and extra large. The optimal outcome for the variable

"cr" has been achieved at approximately 25, while the corresponding processing time is approximately 0.25. The work does not provide an analysis of the practical use of the presented models in real-world contexts.

Another study was presented by Dhiah et al [5]This research has examined the problem of network congestion and bottlenecks resulting from the substantial XML overhead size of medical web services utilized in hospitals that rely on cloud-based web services. To address this problem, a compression-based aggregation model By reducing message sizes is proposed to enhance the effectiveness of web services. Fixed-Length and Huffman are the two compression methods. Compression ratio cr is the primary assessment statistic used in this work. For evaluation, 62 real medical Web XML messages are applied, split into two sets of 31 messages each. In comparing the proposed model with other models, the best results have been achieved, starting from 11.8 to 16.7 for Huffman encoding. It is clear to note that the evaluation of the proposed model was conducted on a specific set of medical web messages and may not be suitable for all types of web services.

Nassima Haroune-Belkacema et al. [6]discussed the use of Web services for exchanging data over the Internet, where SOAP is the main communication protocol in Web services. Consequently, the large size and number of XML based on that protocol leads to high latency and bottlenecks on the Internet. Moreover, that research has investigated the shortcomings of the existing techniques for clustering and compression of these messages. The Smca approach is applied to improve grouping XML data of the same paths in one container. This is achieved by reducing the number and size of messages sharing the same destination and then compressing them in a more efficient manner. In order to assess the effectiveness of the suggested model, three metrics are utilized, namely compression ratio (CR), compression time (CT), and decompression time.

Furthermore, the superior outcomes have been juxtaposed with alternative methodologies. Empirically, a sample size of 160 messages is utilized for the purpose of evaluation. The entities have been classified into four categories according to their size: ((small-medium-large-and, very large)). The proposed technique, known as SMCA, has demonstrated significant gains in channel reliability (CR) across various message sizes. Specifically, when applied to datasets of different scales, SMCA has produced CR enhancements of around 42.40%, 56.77%, 76%, and 77.51% for small, medium, big, and very large messages, respectively. It is crucial to acknowledge that the suggested technique is evaluated on a restricted dataset, and its performance may exhibit variability when applied to different datasets. Moreover, the suggested approach may require supplementary computational capacity and resources for executing various operations, a potential challenge in some settings.

The issue of inefficient performance of soap in environments with a large number of transactions has beeb addressed by Khoii Anh et al [24]. ((SMP)) has been designed to reduce network traffic and optimize traffic size by aggregating similarity messages and sending compact messages. Technically, network traffic and the average response were used to evaluate that model. The FIFA World Cup 2006 dataset was used in the experiment conducted in that paper. The best obtained result is about 47 %, which is equivalent to 2.13 CR.That protocol's additional processing time results in increasing response time. Moreover, more multicast group messages may lead to an increase in the SMTP message header, resulting in transmission delay.

With the aim of reducing network traffic, Kennedy Mutange Senagi et al [25] Have discussed the problem of performance disability caused by using XML documents in high-performance applications that process large amounts of data. Significantly, XML documents used for exchanging structure data are verbose inside and can negatively impact performance. This research proposes an aggregation architecture solution that involves (client-side caching, simple database queries on the server side, and gzip compression technique) to optimize soap performance. This model has been analyzed and evaluated using five main metrics: (compression percentage to ratio- time to transfer SOAP messages- time to process SOAP messages-Round, Trip Time (RTT), and throughput). The best-obtained result of the average Cr percentage observed was 67.01%. In addition, transfer time is improved. The result was conducted only on a disadvantaged network with 10 Mbps bandwidth. The proposed technique may not be effective for all types of SOAP-based applications.

Abdelmalek Habi et al [26] discussed the problem of diffusion and scattering of responses in existing XML information retrieval systems to a given query. The paper highlights the point of Aggregation in solving this problem, where the aggregating search model has replaced the place of the traditional information retrieval method, which produces an ordered list of responses, with the aim of aggregating the most extensive and relevant responses into a single aggregated document. In order to solve this problem, the document suggests using an aggregation approach. Authors have proposed a search model that utilizes the Top-k Approximate Sub-tree Matching (TASM) algorithm to identify the most relevant subtrees in XML documents. Furthermore, they presented an aggregation model that generates a single response document containing the most relevant elements. Two metrics have been used for testing

the proposed model: Recall and precision. Used two real-world datasets in XML format. First of all (DBLP1 ) is a version of bibliographic information on major computer science journals and proceedings. In addition, IMDB2 is a second dataset that represents the Internet Movie Database. The TASM aggregation model shows a high recall for queries, indicating that it retrieves a large amount of relevant information compared to other algorithms. It also demonstrates a high precision, eliminating redundancies and returning only the relevant elements. The evaluation of the proposed method is limited to only two real-world datasets, which may not fully act the diversity and complexity of( XML documents) in different domains.

## 4. Clustering for web aggregation

Web service clustering for the purpose of Aggregation is considered one of the best and most effective solutions for improving the problems of network traffic. This is caused by the drawbacks of XML, which is adopted by web services in its structure.

Al-Shammary et al [27]Have discussed the congestion and bottlenecks caused by web services. Moreover, it takes into account the similarity of a message and how that affects part of the solution to the problem of network congestion. In order to effectively tackle this issue, The concept of fractal self-similarity has been proposed as a computational method for determining the similarity of soap messages. The compression-based aggregation technique is employed following the clustering of messages in order to access the maximum compression ratio for messages. CR and CT methods were employed to test that model. This proposal utilizes 160 messages and is based on the stock quotation Web Service Description Language (WSDL) available at the official website of the World Wide Web Consortium (W3C), http://www.w3.org. The most favorable outcome was achieved for the variable CR, with a range of values from 3.92 to 21.70, accompanied by an average time of around 15.6249 milliseconds. This method has been found to state superiority over the k-means and k-means and (PCA) methods in terms of clustering time when applied to small databases. However, it is important to note that the technique has limitations when used in large databases.

The issue of network congestion is caused by huge requests and large size of soap messages. In addition to highlighting the problem of Aggregation of soap messages at one node as a solution to that traffic problem at different nodes has been discussed BY Dhiah Al-Shammarya et al [28]. A distributed aggregation model has been suggested for the Aggregation of XML messages from several resources sharing the same path. Moreover, a fast fractal similarity-based clustering technique for soap XML was applied. Two measures are used to experiment with their proposed model: processing time and compression ratio. The evaluation of this model involved the utilization of 1800 messages, which were categorized into two sets. The first consisted of 600 messages, which were utilized for the purpose of aggregating accumulation. The second group comprised 1200 messages, which served as a foundation for incorporating newly arrived messages from the network's nodes as they traversed their respective paths. The method under review has demonstrated notable levels of success (3.63, 7.40, 15.42, and 20.12) in terms of AV-Cr, as compared to alternative approaches such as the Fractal methodology and other methodologies.

Another study presented by Nawars et al [8] has examined the challenges associated with web services. Particularly concerning the drawbacks inherited from XML, which lead to the generation of substantial payloads in web messages transmitted across the Internet. Consequently, this results in heightened network traffic. Furthermore, the increased demand for web services as a means of information exchange around the world has contributed to network congestion and performance bottlenecks, resulting in significant delays or complete service interruptions. In simpler terms, SOAP messages inherently possess a considerable size, necessitating a larger bandwidth allocation for both web service requests and responses. Addressing this issue would involve employing clustering and subsequent Aggregation of similar web services into a compressed message, which holds the potential for achieving reductions in network traffic. In this paper, a novel model, static Hilbert clustering, is introduced and designed for clustering web services based on convex set similarity. Mathematically, the proposed model computes similarity metrics among messages, subsequently clustering them based on higher similarity values. Following the clustering process, each cluster is aggregated into a more concise message. (ACR) and average (CT) are employed as a means of evaluating the model. The dataset was constructed utilizing the Web Service Description Language (WSDL). It comprises 160 messages categorized into four distinct groups based on their sizes, namely small, medium, large, and very large, with sizes spanning the range of 140 to 5,500 bytes. The optimal results achievable with this model for the small, medium, large, and very large message groups are as follows: 3.51, 7.98, 16.62, and 20.22, respectively. The proposed clustering forces the data to be distributed completely in static, which may reduce the model's efficiency.

D.alshammary et al. [12] Have discussed the problem of the network performance latency caused by the highly duplicated information in XML-based online messages, which causes congestion and bottlenecks. The large size of XML messages leads to slow network response times and delays in user services. In order to aggregate similar

messages and improve network traffic, the research suggests a clustering model based on Jaccard coefficients to solve this issue. The performance of the suggested model is compared to that of K-Means and K-means plus Principle Component Analysis (PCA). Two measures are used to test this method: Average compression ratio and clustering time. In that model, the dataset consists of messages categorized by their sizes: small, medium, large, and very large. These messages are allocated to different groups based on their sizes. For instance, smaller messages containing around 140 bytes are assigned to a specific small group, while larger messages, such as those around 53 KB, are allocated to a distinct group categorized as very large. The best results are linked to the initial message sizes, that achieved compression ratios of approximately 3.75, 8.04, 16.51, and 20.26 for small, medium, large, and very large messages, respectively. In addition, processing time (15,16,17 and 19). This study does not provide any comparison of the proposed model's performance on different types of XML messages or datasets.

The performance limitations and congestion within Cloud Web services, resulting from the encoding of XML messages that override the actual payloads, Have been solved by Al-Shammary et al [29]]. Performance challenges are observed in SOAP messages of Cloud Web services due to the larger size of XML messages. The paper presented a Fractal clustering approach aimed at calculating the Fractal clustering similarity of SOAP messages, facilitating the Aggregation of SOAP messages to decrease their size. Additionally, the paper proposed two fast Fractal clustering models designed to minimize the time needed for clustering. These introduced fast Fractal models surpass both the traditional Fractal model and the K-means and PCA combined with K-means models in terms of both processing time and reduction in the size of SOAP messages. Experimentally, the major evaluation metrics for the proposed model include average compression ratios (CR) and average processing time. The dataset used in this study comprises 6000 real soap messages, which have been categorized into groups based on their size range: small_medium_large and very large. Each category contains an equal number of 1500 messages. The most extensively studied outcomes of cognitive restructuring (CR) include the values of 3.73, 7.40, 15.42, and 20.12, which have been observed across all categories of messages in the sample.

Ahmed Mohammed Abbas et al. [13]Have examined the problem of bottlenecks for XML Web services that result from the many client requests and the huge size and verbosity of XML Web messages. This study introduces a novel and fast dynamic clustering-based aggregation model for XML Web messages to enhance the performance of the web in the process of exchanging information between clients and servers over the Internet. Technically, (the term Frequency-Inverse Document Frequency ) (TF-IDF) and Euclidean Distance approaches have been used to calculate the high degree of similarity between SOAP messages. Dynamic fractal and VSM approaches were evaluated within AV.CR.The WSDL from http://www.w3.org was used to create the data. It contains 160 messages, which are distributed to groups. There are fourteen messages in each group. For all of the used datasets, the best values for Av. CR were ( 2.941,8.385, 17.36,19.89), briefly for all those categories. This method does not provide any analysis or discussion on the scalability of the proposed method for larger datasets or real-world applications.

## 5. other web services clustering models

Many studies have suggested clustering of XML based Web communications to help with advance Web applications, including information retrieval, data integration, web message structure analysis, and document categorization.

Maciej Piernik et al [30]discussed the popularity of XML documents and the large amounts that are produced every day. Moreover, the paper concentrates on the need to mine data for XML documents. XML clustering is one of the most important XML mining tasks. The paper discussed the challenges of that clustering approach and therefore proposed a new one called (XPattern) based on pattern along with the (PathX). Precision metrics have been used for evaluation. Real and synthetic datasets have been used. Real chosen from the SIGMOD Record and INEX. Synthetic datasets used Heterogeneous (het) and Homogeneous (hom). The best results obtained were 0.66 for the precision metric. Finally, there is no clear evidence that the proposed model is the best outperformance.

Another study by Gianni Costa et al. [31]has discussed the clustering of XML data as a vital issue for unsupervised analysis. It has potential for practical applications like query processing, information extraction, and browsing in addition to web mining. Moreover, XML security is currently considered an essential requirement for presenting system information. It requires using XML clustering tools, which can help separate the input XML documents (or messages) into different categories of beneficial interactions and harmful cyberattacks. The study proposes the utilization of n-grams as distinguishing attributes in XML documents for the purpose of grouping or clustering them according to similarities in both their content as well as their structure. In essence, the utilization of n-grams facilitates

the identification of shared patterns and thematic material among papers that exhibit comparable arrangements of constituents. Three machine-learning techniques were applied (XC-NMF, XCo-Clust, and XPart). Evaluation is achieved by computing the Micro-averaged Purity and Macro-averaged Purity. Wikipedia and Sigmod datasets have been applied to test the performance of that method. Micro average Purity of (0.58,0.91)has been obtained as the best result for xco-clust, moreover (0.75,0.95) of macro Purity for Wikipedia and Sigmod dataset. However, it is important to note that the effectiveness of the three machine learning approaches used in the method may vary, depending on the length of contextualized n-grams. Moreover, the proposed method may not be suitable for all types of XML documents.

S.Sahunthala et al [32]]This paper describes the need for clustering XML data in (real-time )web applications. In addition, it discusses the importance of internet data and the role of markup languages, particularly XML, in processing web data. This article focused on the limitations of existing clustering methods for analyzing XML data and then suggested the Improved Fuzzy Clustering Method (I-FCM) as a solution to improve clustering performance. Parameters such as time, precision, recall, entropy, and Purity were used to compare the model's efficacy with existing (WFCM), LSI, and K-means clustering approaches. Used one dataset, namely LF-AMAZON.The best results achieved were 0.89, 0.09, 0.73, 0.114, and 1 for precision, recall, entropy, and Purity sequentially. The proposed Improved Fuzzy Clustering Method (I-FCM) is compared only with limited methods. There may be other methods that provide better results. The experimental evaluation worked only on one dataset. It may perform differently on other datasets.

The issue of growing a large amount of XML data and its advantages currently. As a result, clustering XML documents has become a popular topic in XML mining and has been discussed by Di Zhao et al [33]. In addition, discussed the lack of fast clustering algorithms for XML data structures such as k-mean In addition to its ineffectiveness for large-scale XML data. A new XML clustering algorithm called CO-LSPX is proposed, which is based on the cluster core LSPX, with the aim of increasing the efficiency of clustering by reducing time consumption and improving the sensitivity of input data to the order clustering algorithms. Moreover, it has provided as a solution for large-scale data. TIME CONTRAST and accuracy have been used for testing and comparing with (TED+K-medoids) and the PBClustering algorithm. Two datasets were used to assess and test this model. Initial supply from INEX (Movie data set). Shakespeare data, known as (Sigmod) and SIGMOD, supply the second. The proposed model has resulted in (1.8, 0.7) clustering time. In addition, the average accuracy of CO-LSPX and TED+k-medoids are the same but higher than that of PBClustering. Moreover, the accuracy of (CO-LSPX) does not change when the input sequence is changed. The proposed algorithm is based on the LSPX model, which may not be suitable for all types of XML data structures.

Gianni Costa et al [34]This study has dealt with XML document clustering as a main problem. This covers the difficulties of capturing content and structural similarity in XML documents, detecting structural and textual data parallels, and determining semantic connections between textual data and structure labels. The paper suggests two methods for resolving this problem. The first method uses clustering algorithms to divide XML document semantic representations according to latent topic modeling. The MUESLI model is applied to implement this strategy. The second method uses the PAELLA model to integrate topic modeling and XML document clustering into one process. The model links a latent cluster-membership random variable to each XML document and blends MUESLI with a Bayesian probabilistic model. Two measures are used for evaluation ( macro-averaged and micro-averaged Purity). The two real-world standard XML corpora used in this study are Sigmod and Wikipedia. The online digital encyclopedia's 47,397 articles make up the Wikipedia corpus. The SIGMOD Record problems are represented by 140 XML pages in the Sigmod corpus. The paper demonstrates that the proposed model, MUESLI, in addition to PAELLA, outperforms other models. The evaluation of the suggested methods is restricted to two references: XML corpora, Sigmod, and Wikipedia. In order to determine how well the models generalize, it would be advantageous to evaluate their performance on a larger variety of XML datasets.

In another study by Guo Yongminget et [35]], the clustering of XML documents is the issue covered in the paper. The authors illustrate that traditional clustering algorithms ignore the content of XML documents and instead focus on the structural similarities between them. They suggest a technique for efficiently clustering XML documents that integrates both structure and content. Taking into account both structure and content, the authors of this study developed a novel method called the Extended Vector Space Model (EVSM) to compare the similarities of XML documents. The research paper presents two variants of the basic extended vector space. The performance of that model is evaluated by applying standard criteria such as Purity and entropy. The datasets utilized in experiments are the (INEX IEEE) and (Wikipedia collection). The IEEE corpus is better than the Wikipedia collection. Where results refer to The relatively higher Purity ( 73%) and lower entropy (0.146), the study does not discuss the scalability of the proposed method for clustering another-scale XML document collections.

## 6. Analysis and Evaluation

In this study, 25 models for web service aggregation, compression, and clustering are provided. The analysis strategy used the average compression ratio (AV. CR) values extracted from most publications to assess the models' efficacy. All these have been methodically divided into different categories, namely, Web Services Clustering approaches for Aggregation. This category encompasses six research papers and other Web Services Clustering approaches. Within this category, six research papers have been included: Aggregation Approaches A total of six research papers have been classified under this grouping, and Compression Approaches: Seven research papers fall under the category of compression techniques. This organized categorization makes it easier to evaluate and analyze the information in a structured manner according to its many domains and aims.

Table 1 presents the method (Novel compression model for structural data) that emerges as the winner, surpassing rival strategies in all data categories. Furthermore, the approaches labeled IRMC, D2D, and ARES have demonstrated the least favorable CR results in Table 2. It is evident that, as compared to the other models, the two and one-bit strategies showed more CR values. Conversely, it is clear that the remaining models recorded comparatively lower compression ratios, including SMCA, Binary Tree Clustering, AMP, and others. Table 3 presents the optimal results of Average Compression Ratios (AV. CR) for aggregation approaches employing web services clustering techniques. In terms of AV and CR, it has been empirically found that the dynamic clustering technique based on fractal similarity measures performs better. Additionally, Table 4 provides an overview of the superior outcomes achieved by Web Services Clustering approaches utilized for diverse purposes, employing multiple measures such as precision, recall, micro average Purity, macro average Purity, and entropy.

**Table 1 CR for compression approaches.**

| **Author** | **model** | **C r** | |
|---|---|---|---|
| G.P.TIWARY et al. [14] | Noval- compression model for structure data | small | 10.8 |
| | | medium | 16.5 |
| | | large | 21.45 |
| | | Very large | 22.85 |
| Dhiah Al-Shammary at El [21] | Variable and fixed length coding | Small | 2.037 |
| | | Medium | 3.015 |
| | | Large | 7.8 |
| | | Very large | 13.45 |
| Hejun Xuet al [18] | Iterative reference mapping method (IRMC) | 2.43 | |
| Athanasios Kiourtis et al [19] | new protocol D2D (device to device) | 2.5 | |
| Chengxi Bernad Siew et al [22] | CityGML Schema Aware Compressor | 10 | |
| Hariharan Devarajan at el [23] | (Ares) | 13.85 | |

**Table 2 best CR based aggregation**

| **Authors** | **method** | **Cr** | |
|---|---|---|---|
| l-Shammarya et_al [1] | (Two and One_bit )aggregation approaches | Sm | 3.65 |
| | | Me | 8.04 |
| | | LA | 16.51 |
| | | V. large | 20.26 |
| Dhiah Al-Shammarya et al [5] | Binary tree aggregation model | 16.7 | |

| | | | |
|---|---|---|---|
| N.Haroune-Belkacema et al [6] | (SMCA) | Sm | 1.74 |
| | | Me | 2.3 |
| | | LA | 4.16 |
| | | V. large | 4.44 |
| Phan et_al [24] | (SMP) | 2.13 | |
| K. M. Senagi et al [25] | AggregatING technique depend on Gzip | 3_ | |

**Table 3 Best CR of clustering for w.b aggregation**

| **Author** | **approaches** | **Av_cr** | | | |
|---|---|---|---|---|---|
| | | S_message | M_message | L_message | VL_message |
| D. Al-Shammarya et al [27]] | Dynamic Fractal self-similarity measurement | (3.92) | (7.98) | (16.63) | (21.70) |
| D.Al-Shammarya et al [28] | Distributed Aggregation by Fractal Similarity Model | 3.75 | 7.63 | 15.90 | 20.75 |
| D. Al-Shammarya et_al [12] | Jaccared coefficients | 3.65 | 8.04 | 16.51 | 20.26 |
| D. Al-Shammarya et al [29] | Fast fractal clustering | 3.63 | 7.40 | 15.42 | 20.12 |
| A.M Abbas et_al[13] | Dynamic clustering based on(TF-IDF) | 2.94 | 8.38 | 17.36 | 19.89 |

| | | | | | |
|---|---|---|---|---|---|
| Nawras et al [8] | Static Hilbert clustering | 3.51 | 7.98 | 16.62 | 20.22 |

**Table 4 Best CR of other WS clustering**

| **author** | **Model** | **Metrics** | |
|---|---|---|---|
| S.Sahunthala et al [32] | Fuzzy Clustering Method (I-FCM) | precision | 0.89 |
| | | recall | 0.09 |
| Di Zhao et al [33] | CO-LSPX | Clustering time =1.8, 0.7 | |
| Gianni Costa et al [31] | XCo-Clust | Micro average purity=0.58 Macro average purity=0.91 | |
| Maciej Piernik et al. [30] | XPattern | 0.66 | |
| Gianni Costa at el [34] | PAELLA<br>MUESLI | -<br>- | |
| Guo Yongming at El [35] | Extended Vector Space Model (EVSM | Purity= 73% Entropy= 0.146) | |

This paper has employed a graphical bar diagram to represent the data shown in Tables 1, 2, and 3 in order to obtain a thorough understanding of the effectiveness of all models. A comparison of the (Av. CR) Outcome of

compression methods used for traffic reduction is shown in Figure 1. The AV. CR results for the Aggregation and clustering for web services aggregation models are shown in Figures 2 and 3, respectively. Moreover, the average CR for each recognized group is calculated. Figure 4 provides an overview of the average CR results encompassing all clustering for web service aggregation, Aggregation, and compression models. However, it is interesting that clustering techniques have outperformed others, averaging up to (14.12) CR. Moreover, compression methods came in second place with an average compression ratio of (10.5) CR. Aggregation methods, on the other hand, have shown the lowest rate with a CR of (6.02).

In sum, based on the identified gaps in this study, future research should focus on developing more efficient clustering algorithms that can handle dynamic web service environments. Innovative compression techniques that can further reduce XML message sizes without compromising data integrity are also needed. Moreover, exploring hybrid approaches that combine clustering and compression methods could offer significant improvements in network efficiency.

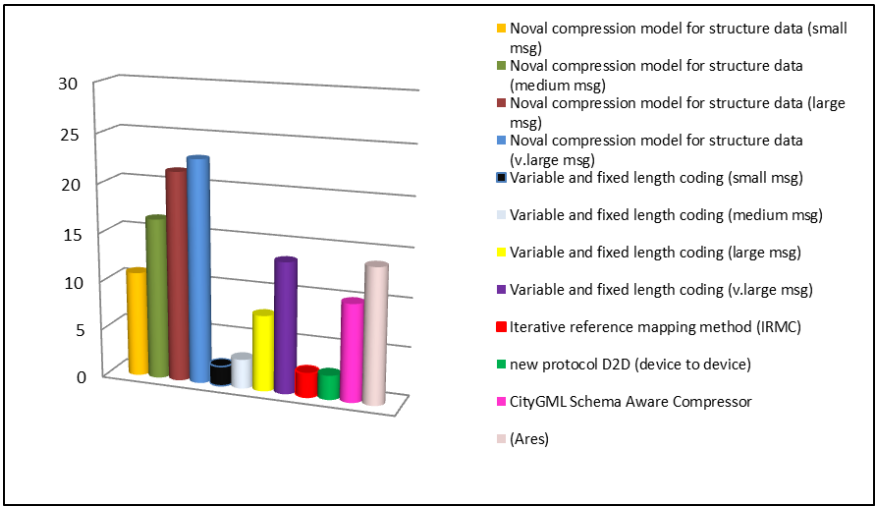


**Figure1 C.R for compression models**

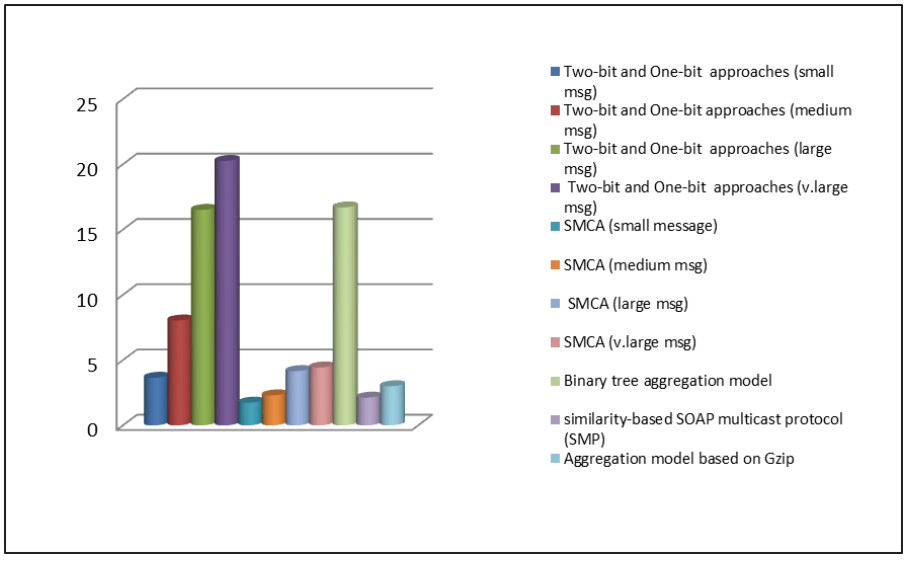


**Figure 2 CR for aggregation modeles**

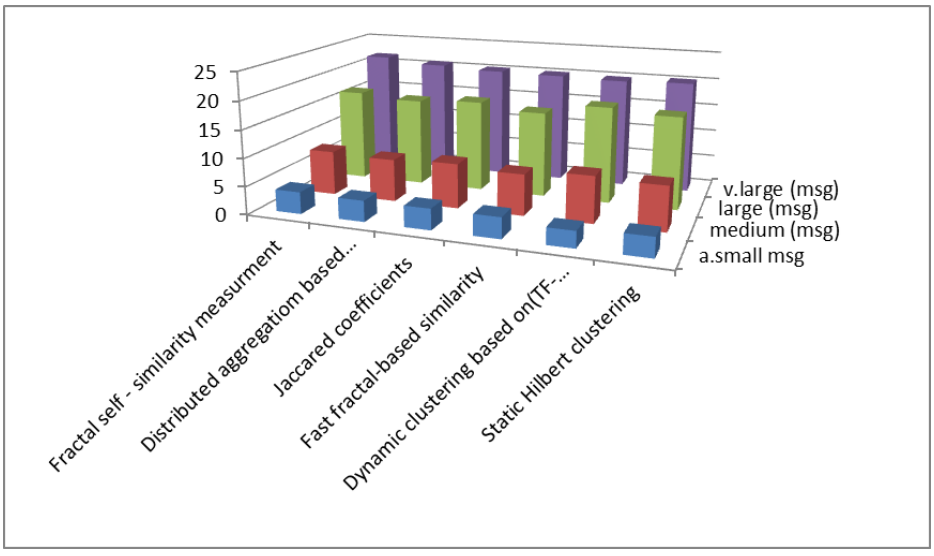


**Figure 3 CR of clustering for web aggregation**

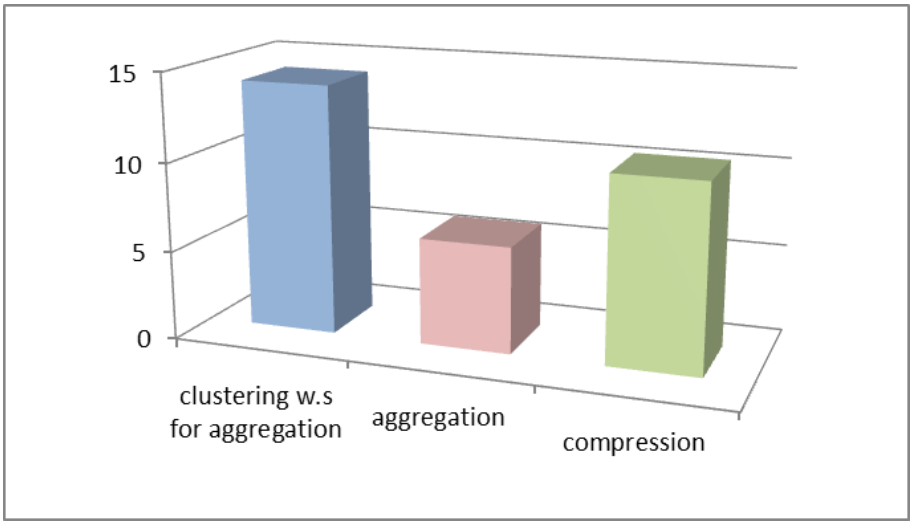


**Figure 4 Outcomes of compression rate averages for various techniques**

## 7. conclusion

This study has encompassed a total of twenty-five models regarding clustering compression and aggregation approaches specifically designed for XML documents. From a technical standpoint, the methods can be classified into four distinct categories: compression techniques, aggregation techniques, clustering algorithms for web services aggregation, and clustering algorithms for another purpose. (Compression ratios) are obtained and utilized from the existing literature on the offered models to assess the effectiveness of the strategies. The (CR) is obtained and subsequently compared for each group. In the context of traffic reduction, it has been observed that (dynamic clustering) utilizing fractal similarity measures exhibits superior performance compared to alternative clustering techniques. The Two-bit and One-bit approaches demonstrated superior classification rate (CR) performance compared to previous aggregation methods. In addition, the novel compression model designed for structured data has demonstrated superior performance compared to previous compression models regarding compression ratio. When comparing all model groups, it was found that the clustering techniques for Aggregation had the highest CR, as indicated by the average CR.

## References


[1] D. Al-Shammary and I. Khalil, "Redundancy-aware SOAP messages compression and aggregation for enhanced performance," *J. Netw. Comput. Appl.*, vol. 35, no. 1, pp. 365–381, 2012.

[2] A. Barkat, O. Kazar, and I. Seddiki, "Framework for web service composition based on QoS in the multi cloud environment," *Int. J. Inf. Technol.*, vol. 13, no. 2, pp. 459–467, 2021.

[3] S. Sagayaraj and M. Santhoshkumar, "Heterogeneous ensemble learning method for personalized semantic web service recommendation," *Int. J. Inf. Technol.*, vol. 12, no. 3, pp. 983–994, 2020.

[4] H. K. Sowmya and R. J. Anandhi, "An efficient and scalable dynamic session identification framework for web usage mining," *Int. J. Inf. Technol.*, vol. 14, no. 3, pp. 1515–1523, 2022.

[5] D. Al-Shammary and I. Khalil, "Compression-based aggregation model for medical web services," in *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, IEEE, 2010, pp. 6174–6177. doi: 10.1109/IEMBS.2010.5627759.

[6] N. Haroune-Belkacem, F. Semchedine, A. Al-Shammari, and D. Aissani, "SMCA: An efficient SOAP messages compression and aggregation technique for improving web services performance," *J. Parallel Distrib. Comput.*, vol. 133, pp. 149–158, 2019.

[7] N. Agarwal, G. Sikka, and L. K. Awasthi, "Evaluation of web service clustering using Dirichlet Multinomial Mixture model based approach for Dimensionality Reduction in service representation," *Inf. Process. Manag.*, vol. 57, no. 4, p. 102238, 2020.

[8] N. A. Al-Musawi and D. Al-Shammary, "Static Hilbert convex set clustering for web services aggregation," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 32, no. 1, pp. 372–380, 2023.

[9] Y. Liu and Z. Hong, "Mapping XML to RDF: An algorithm based on element classification and aggregation," in *Journal of Physics: Conference Series*, IOP Publishing, 2021, p. 12012.

[10] N. A. Al-Musawi and D. Al-Shammary, "Dynamic Hilbert clustering based on convex set for web services aggregation," *Int. J. Electr. Comput. Eng.*, vol. 13, no. 6, pp. 6654–6662, 2023.

[11] C. Ben Njima, C. G. Guegan, Y. Gamha, and L. Ben Romdhane, "Web Service Composition in Mobile Environment: A survey of Techniques," *IEEE Trans. Serv. Comput.*, 2024.

[12] D. Al-Shammary, "Jaccard Coefficients based Clustering of XML Web Messages for Network Traffic Aggregation," *J. Al-Qadisiyah Comput. Sci. Math.*, vol. 11, no. 2, pp. 82–91, 2019.

[13] A. M. Abbas, A. A. Bakar, and M. Z. Ahmad, "Fast dynamic clustering SOAP messages based compression and aggregation model for enhanced performance of Web services," *J. Netw. Comput. Appl.*, vol. 41, pp. 80–88, 2014.

[14] G. P. Tiwary, E. Stroulia, and A. Srivastava, "Compression of xml and json api responses," *IEEE Access*, vol. 9, pp. 57426–57439, 2021.

[15] M. Ericsson, "The effects of xml compression on soap performance," *World Wide Web*, vol. 10, pp. 279–307, 2007.

[16] C. Werner and C. Buschmann, "Compressing SOAP messages by using differential encoding," in *Proceedings. IEEE International Conference on Web Services, 2004.*, IEEE, 2004, pp. 540–547.

[17] J. C. Estrella, M. J. Santana, R. H. C. Santana, and F. J. Monaco, "Real-time compression of soap messages in a soa environment," in *Proceedings of the 26th annual ACM international conference on Design of communication*, 2008, pp. 163–168.

[18] H. Xu, J. I. Kim, and J. Chen, "An iterative reference mapping approach for BIM IFCXML classified content compression," *Adv. Eng. Informatics*, vol. 54, no. January, p. 101788, 2022, doi: 10.1016/j.aei.2022.101788.

[19] A. Kiourtis, A. Mavrogiorgou, and D. Kyriazis, "Health information exchange through a Device-to-Device protocol supporting lossless encoding and decoding," *J. Biomed. Inform.*, vol. 134, no. September, p. 104199, 2022, doi: 10.1016/j.jbi.2022.104199.

[20] D. Hucke, M. Lohrey, and L. S. Benkner, "Entropy Bounds for Grammar-Based Tree Compressors," *IEEE Trans. Inf. Theory*, vol. 67, no. 11, pp. 7596–7615, 2021, doi: 10.1109/TIT.2021.3112676.

[21] D. Al-Shammary and I. Khalil, "SOAP web services compression using variable and fixed length coding," in *2010 Ninth IEEE International Symposium on Network Computing and Applications*, IEEE, 2010, pp. 84–91.

[22] C. B. Siew and P. Kumar, "CitySAC: a query-able cityGML compression system," *Smart Cities*, vol. 2, no. 1, pp. 106–117, 2019.

[23] H. Devarajan, A. Kougkas, and X.-H. Sun, "An intelligent, adaptive, and flexible data compression framework," in *2019 19th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID)*, IEEE, 2019, pp. 82–91.

[24] K. A. Phan, Z. Tari, and P. Bertok, "Optimizing Web services performance by using similarity-based multicast protocol," *Proc. ECOWS 2006 Fourth Eur. Conf. Web Serv.*, pp. 119–126, 2006, doi: 10.1109/ECOWS.2006.29.

[25] K. M. Senagi, G. Okeyo, W. Cheruiyot, and M. Kimwele, "An aggregated technique for optimization of SOAP performance in communication in Web services," *Serv. Oriented Comput. Appl.*, vol. 10, no. 3, pp. 273–278, 2016, doi: 10.1007/s11761-015-0186-x.

[26] A. Habi, B. Effantin, and H. Kheddouci, "Search and aggregation in XML documents," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10438 LNCS, pp. 290–304, 2017, doi: 10.1007/978-3-319-64468-4_22.

[27] D. Al-Shammary, I. Khalil, Z. Tari, and A. Y. Zomaya, "Fractal self-similarity measurements based clustering technique for SOAP Web messages," *J. Parallel Distrib. Comput.*, vol. 73, no. 5, pp. 664–676, 2013.

[28] D. Al-Shammary, I. Khalil, and Z. Tari, "A distributed aggregation and fast fractal clustering approach for SOAP traffic," *J. Netw. Comput.*

*Appl.*, vol. 41, pp. 1–14, 2014.

[29] D. Al-Shammary, I. Khalil, and L. E. George, "Clustering SOAP Web services on internet computing using fast fractals," in *2011 IEEE 10th International Symposium on Network Computing and Applications*, IEEE, 2011, pp. 366–371.

[30] M. Piernik, D. Brzezinski, and T. Morzy, "Clustering XML documents by patterns," *Knowl. Inf. Syst.*, vol. 46, no. 1, pp. 185–212, 2016, doi: 10.1007/s10115-015-0820-0.

[31] G. Costa and R. Ortale, "Machine learning techniques for XML (co-)clustering by structure-constrained phrases," *Inf. Retr. J.*, vol. 21, no. 1, pp. 24–55, 2018, doi: 10.1007/s10791-017-9314-x.

[32] S. Sahunthala, A. Geetha, and L. Parthiban, "An Improved Fuzzy Clustering Method for Analyzing Clustering Quality in XML Web Data," *4th Int. Conf. Inven. Res. Comput. Appl. ICIRCA 2022 - Proc.*, no. Icirca, pp. 1329–1335, 2022, doi: 10.1109/ICIRCA54612.2022.9985517.

[33] D. Zhao, H. Fu, H. Ren, M. Wei, and J. Chu, "on Cluster Core And LSPX," pp. 1027–1032, 2017.

[34] G. Costa and R. Ortale, "Mining cluster patterns in XML corpora via latent topic models of content and structure," in *Advances in Knowledge Discovery and Data Mining: 23rd Pacific-Asia Conference, PAKDD 2019, Macau, China, April 14-17, 2019, Proceedings, Part III 23*, Springer, 2019, pp. 237–248.

[35] G. Yongming, C. Dehua, and L. Jiajin, "Clustering XML documents by combining content and structure," in *2008 International Symposium on Information Science and Engineering*, IEEE, 2008, pp. 583–587.