International Journal of Electrical and Computer Engineering (IJECE)
Vol. 13, No. 6, December 2023, pp. 6654~6662
ISSN: 2088-8708, DOI: 10.11591/ijece.v13i6.pp6654-6662

Dynamic Hilbert clustering based on convex set for web services aggregation

Nawras A. Al-Musawi, Dhiah Al-Shammary
College of Computer Science and Information Technology, University of Al-Qadisiyah, Diwaniyah, Iraq

Article Info

Article history:

Received Oct 13, 2022
Revised Mar 18, 2023
Accepted Apr 3, 2023

Keywords:

Aggregation
Clustering
Hilbert
Message
Web services

ABSTRACT

In recent years, web services run by big corporations and different application-specific data centers have all been embraced by several companies worldwide. Web services provide several benefits when compared to other communication technologies. However, it still suffers from congestion and bottlenecks as well as a significant delay due to the tremendous load caused by a large number of web service requests from end users. Clustering and then aggregating similar web services as one compressed message can potentially achieve network traffic reduction. This paper proposes a dynamic Hilbert clustering as a new model for clustering web services based on convex set similarity. Mathematically, the suggested models compute the degree of similarity between simple object access protocol (SOAP) messages and then cluster them into groups with high similarity. Next, each cluster is aggregated as a compact message that is finally encoded by fixed-length or Huffman. The experiment results have shown the suggested model performs better than the conventional clustering techniques in terms of compression ratio. The suggested model has produced the best results, reaching up to 15 with fixed-length and up to 20 with Huffman

This is an open access article under the CC BY-SA license.

Corresponding Author:

Nawras A. Al-Musawi
College of Computer Science and Information Technology, University of Al-Qadisiyah
Diwaniyah, Iraq
Email: nawras.kadhim@qu.edu.iq

1. INTRODUCTION

Web services can be defined as web application that communicates with other systems using standard Internet mechanisms such as hypertext transfer protocol (HTTP) and extensible markup language (XML) in order to provide the required service to the user [1]–[3]. Web services are competitive technologies that allow e-business and e-commerce by offering clients quick, efficient, and dependable services through the internet [4]. Web services are a kind of simple object access (SOA)-based distributed computing paradigm that can be accessed through HTTP. Web servers offer dynamic replies for all requests made via the internet [5]–[7]. Web service architecture refers to the software system design for supporting interoperable communication from machine to machine via a network [1], [8]. Web services have become one of the most important areas of technological development in recent years [4], [9]. Web services represent the most usable communication protocol over the net. Web services communication is built on standard technologies such as simple object access protocol (SOAP), web service description language (WSDL), and universal description discovery and integration (UDDI) [10], [11]. The SOAP protocol is a widely-used communication protocol based on XML Web service [12], [13]. SOAP is a protocol for allowing different distributed computing systems to communicate with each other [14]. Organizations are increasingly using web services due to the technological advantages of web services that provide interoperability and dynamic scalability [12], [15].

The XML-based encoding of web messages has harmed the network's efficiency due to the high bandwidth that their excessive repetition of information has caused [16]. Because of the large amount of XML data being sent across the network, consumers have suffered from bottlenecks and congestion [17]. Most of the time, Network latency is a major issue that often causes service delays for end users [18]. Particularly, SOAP messages are inherently large, necessitating greater bandwidth for web service requests and responses [2], [19]. This fact has driven researchers to create solutions that minimize loads on networks and increase the speed at which services may be provided. One of the newest suggested improvements is a web-based messages aggregator, which works to combine several messages into a single message by removing duplicate content [20], [21]. However, the aggregation's efficacy depends on the degree of similarity between aggregated messages. To provide high aggregation outcomes, a similarity measurement model is necessary. This demand would be supported by effective clustering, which would as well as enhance network-based web services to endpoints.

Technically, the proposed model has provided several major contributions in regard to the web service performance over the network. Firstly, it would improve web-based communication over the network by significantly reducing the required network volume. This fact would solve the network congestion and bottleneck significantly. Secondly, efficient web messages clustering has been achieved based on Hilbert similarity measurements. Finally, the proposed model would potentially improve web service response time as a result of the required low processing time.

An efficient evaluation strategy is established and executed by evaluating the effectiveness of the suggested model. This is accomplished by sending the results of its grouped message to the aggregator model developed by [17], [22] and examining its effects on processing times and compression ratios for network traffic reductions. Other clustering models, including k-means and principal component analysis (PCA) coupled with k-means, are treated in the same way since they serve as a standard for the suggested clustering method. Compared to other methods, the Hilbert clustering model has been shown to be superior due to its ability to achieve similar outcomes while using less processing time and generating significantly less network traffic.

The rest of this paper is structured as: section 2 explains the related work. Section 3 describes the proposed model. Experiments and results are discussed in section 4. Finally, section 5 presents the conclusion and future work.

2. RELATED WORK

XML message clustering is essential in web service environments and applications. Machine learning methods have often been used for clustering web services. Many researchers have technically presented different clustering methods for textual and XML documents. The primary focus of the proposed clustering models has been on clustering XML documents. Similarly, another major trend in clustering methods is focused on numerical data and hence cannot be used to text without first being converted to a similar binary form.

Hwang and Gu [23] presented a new approach to grouping XML texts based on the repetitive structure weight in XML texts, treating XML document is viewed as a transaction and its structures are extracted as transactional items. The suggested method focuses on the idea of identifying the large elements included in each XML document and then grouping them together based on their similarities. Evaluation metrics consist of XML tag and data item paths in the XML tree. A basic principle of the suggested method is to allocate new XML documents according to an average of the accumulated frequency of structures in the XML tree. The suggested method is compared to both the hierarchical agglomerative clustering (HAC) and k-means methods to determine its efficacy. The experimental results demonstrate that the suggested method outperforms both HAC and k-means.

The extended vector space model (EVSM) was developed by Yongming *et al.* [24] as a novel model for grouping XML documents according to their structure and content by calculating similarities between them. Additionally, the hierarchical clustering method has been used in this approach. The effectiveness of the suggested clustering approach was measured using two different criteria. Purity and entropy are two of these measures. Their model has been experimentally tested on two datasets (the INEX IEEE corpus and the Wikipedia collection). According to the findings, this approach would result in more efficient clustering.

A dynamic clustering approach for SOAP web messages was implemented by Al-Shammary *et al.* [17] based on fractal similarity measurements. The clustered messages resultant has been formed in a compact form for network traffic reducing. The proposed model has been examined using the compression ratio and processing time. By analyzing the web aggregator's resulting compression ratio (CR), the suggested model has been compared with both k-means and PCA+k-means. The dataset has been utilized for testing and experimentation consist of 160 SOAP messages, divided into four sets of 40 messages each. These messages were created using the WSDL for stock quotes found at [25]. The highest Av.CR values were obtained for the following datasets, in that order: 3.92, 7.98, 16.63, and 21.70. Average processing time for the proposed model takes around 15.6 milliseconds.

3. PROPOSED MODEL

This section explains the complete design for proposed model including Hilbert convex function, web services aggregation and Hilbert clustering model. First, Hilbert convex set as similarity measurements are explained with the mathematical and equations. Moreover, the proposed model includes the implemented techniques for web service aggregation that combine web messages in a compact size based on the Hilbert similarity grouping. Finally, the proposed model has included a new dynamic clustering technique based on Hilbert measurements.

3.1. Hilbert convex set as similarity measurements

In this research, we have proposed Hilbert based on convex set measurements for SOAP messages clustering. We have considered each message as convex (set) and each feature in the message as a point. Technically, convex set measurements are applied to calculate the similarity between two messages. Equation (1) illustrates the Hilbert convex set measurements, including their parameters:

$$Sim\ (x, y) = \frac{\sum_{i=1}^{n}(X_i * Y_i) - Max\ (x)}{\|x\| * \|y\|} \tag{1}$$

$$\|x\| = \sqrt{x_1^2 + x_2^2\ + x_3^2 + \cdots} \tag{2}$$

$$\|y\| = \sqrt{y_1^2 + y_2^2\ + y_3^2 + \cdots} \tag{3}$$

where, *x* and *y* are vectors (*x* and *y* are represented by SOAP messages), *n* is number of attributes in vector, and *Max(x)* is maximum value in vector *x*.

3.2. Web service aggregation

In this study, an aggregation model based on compression [17], [22] is used to examine the performance of the suggested approach. The paradigm of aggregation consists of three phases leading to a compact aggregated message. Initially, XML messages are transformed into a web matrix (in matrix form), and then they are represented as an XML tree. XML trees are then transformed into vectors using either depth-first or breadth-first tree traversal. These text expressions are compressed using either the fixed-length or Huffman lossless compression methods. The suggested model has been formed into two primary approaches: one-bit and two-bit aggregators. Both aggregators provide two versions of the fixed-length and Huffman encoding techniques.

3.3. Dynamic Hilbert clustering model

A novel Hilbert clustering technique is proposed to provide dynamic clustering for web services. Hilbert space and distance measurements are introduced as new mathematical computations for the similarity between XML messages. Hilbert is suggested as a strategy for unsupervised clustering, which groups XML messages in a dynamic manner. The presented approach intends to build clusters with a high similarity degree by grouping messages together dynamically without requiring a fixed number of clusters. The proposed model first converts Web messages into numeric form developed by Al-Shammary *et al.* [17] and Hwang and Gu [23]. Next, Hilbert convex set measurements are achieved on the numeric representations of XML messages. Technically, Hilbert similarity measurements are applied in order to cluster XML messages based on the maximum similarity values. The final Hilbert clusters would be the input to the We aggregators to aggregate each cluster’s messages together. Figure 1 explains the main steps of the proposed Hilbert clustering model.

Practically, dynamic Hilbert clustering begins with constructing an XML tree that maintains a single occurrence in the tree while eliminating duplicate ending tags (closing tags), then employing a term frequency-inverse document frequency (TF-IDF) weighting technique to calculate elements’ frequency for each message. Then the outcomes of this method will be recorded in a vectors matrix of two dimensions. Next, Hilbert convex measurements have been applied to compute the similarity degree between messages. Each (current) message’s Hilbert similarity measurements are computed with all messages and compared, and the message is then assigned to the cluster with the highest similarity value. Eventually, the resulting clusters are constructed in the string form and input into an aggregator tool. This tool combines the contents of each cluster of messages that are similar into a single compressed message. Figure 2 shows a detailed explanation of all steps taken by the dynamic clustering model. Algorithm 1 illustrate this process.

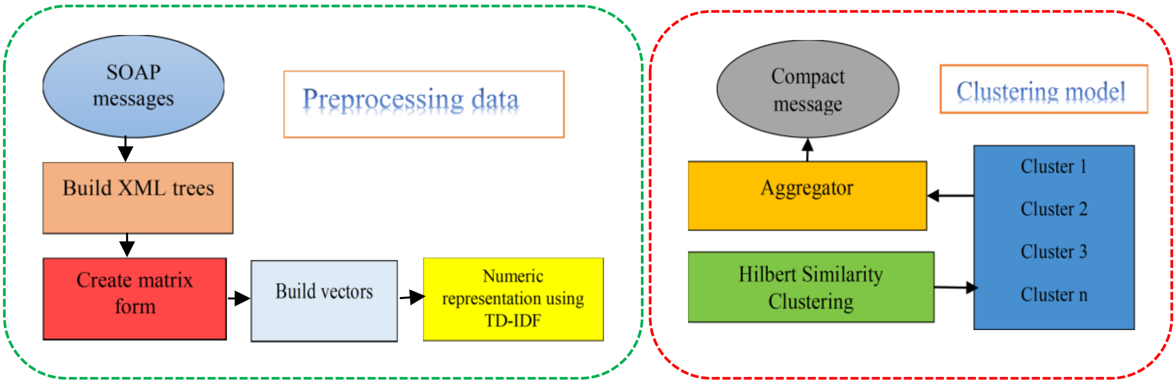


Figure 1. Main steps of Hilbert clustering

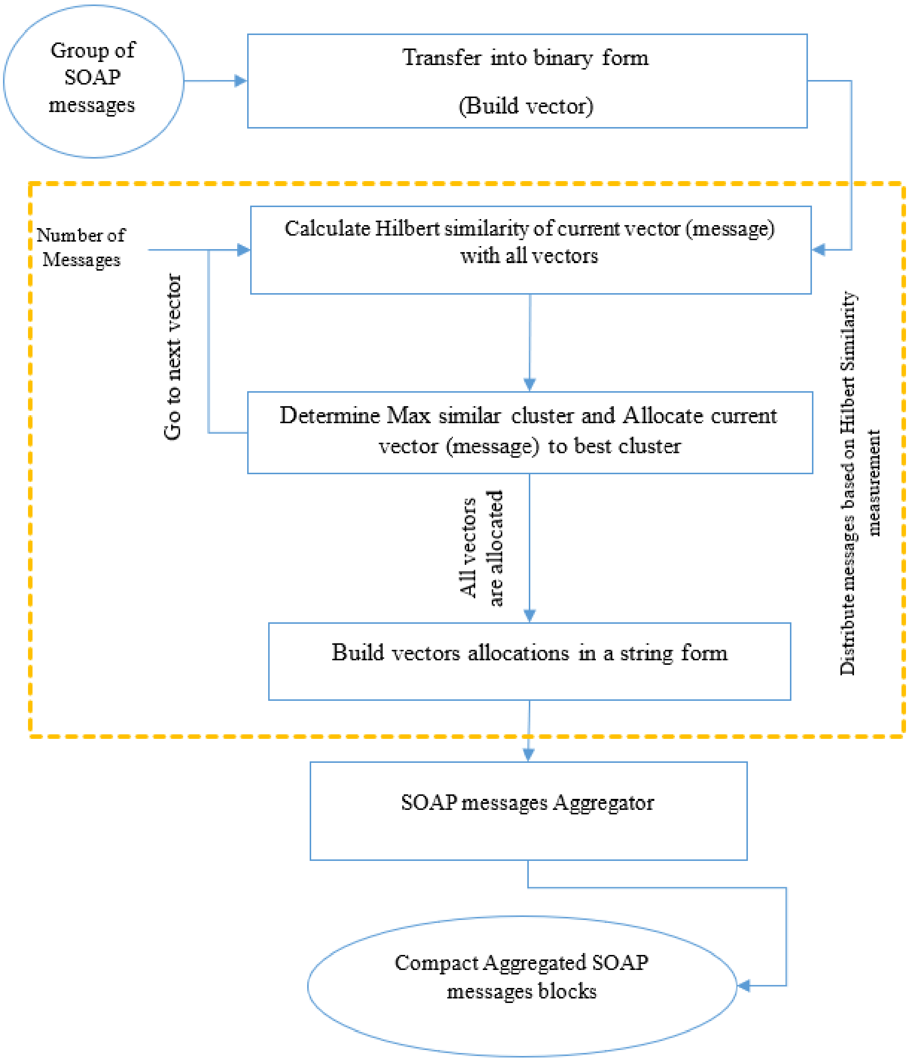

Group of SOAP messages
Transfer into binary form
(Build vector)
Number of Messages
Calculate Hilbert similarity of current vector (message) with all vectors
Go to next vector
Determine Max similar cluster and Allocate current vector (message) to best cluster
All vectors are allocated
Build vectors allocations in a string form
Distribute messages based on Hilbert Similarity measurement
SOAP messages Aggregator
Compact Aggregated SOAP messages blocks

Figure 2. Dynamic Hilbert clustering model

Algorithm 1. Dynamic Hilbert clustering (messages distribution)

```
    //Definitions
Sn  //Number of message in dataset
Cn  //Number of clusters
Vs  //Vector items
Vflag (Sn)  //Array of flag to determine the allocated centers
For $i$=1 to Sn  //compute sum weight and find norm for all messages
   Norm (i)=$\sqrt{\sum_{f=1}^{N} X_f^2}$
End for
For $i$=1 to Vs  //All vector items
    //find Maximum value (Max)
End for
For $i$=1 to Sn  //Compute Hilbert similarity
   If Vflag ($i$)=false then
      For $j$=1 to Sn//All messages
         If ($i$< >$j$) then
             Dot=$\sum_{r=1}^{N}[Wj(r) \times Wi(r)]$
                  HS ($i$)=$\frac{Dot-Max(i)}{Norm(i)\times Norm(j)}$
                        //Find maximum similarity (closest message)
          End if
      End for
      //Add message to the cluster
      //Exclude message
  End if
End for
For $i$=1 to Cn
    //generate string form for all allocation vectors (clusters)
End for
```

4. EXPERIMENTS AND RESULTS

In order to evaluate the performance of the proposed Hilbert clustering model, a wide range of XML message sizes has been considered in the experimental analysis. The dataset was formed using the WSDL at [25]. It contains 160 messages divided into four groups based on their size. Which categories: small, medium, large, and very large, with sizes ranging from 140 to 5,500 bytes. Each group has 40 messages [17]. We have used the compression-based aggregation model [22] to investigate the proposed model's efficiency compared to other standard clustering methods. Vector space model [26] has been applied to the same dataset and compared with the proposed model. The main evaluation metrics include compression ratio, clustering time and compressed size file that are computed by aggregator tool on the resultant clusters. The highest CR achieved by these methods means this is the best clustering. Furthermore, other comparisons are applied with k-means and PCA+k-means from previous studies [17], [27].

In order to achieve potentially large compression ratios on the clustered XML messages, all strategies demonstrated substantial results. Experimentally, the structure of clusters for each group is explained in Table 1. The clusters that formed as a result of the small group distribute different numbers of XML messages (10, 20, 30, and 40) into variety clusters (3, 8, 10, 13), illustrating the dynamic nature of the Hilbert model since the size of the cluster is not predetermined. In the medium group, the number of clusters is 3, 6, 9, and 11, including 10, 20, 30, and 40 messages, respectively. Then again, the large group with 10, 20, 30, and 40 messages distributed into 3, 6, 6, and 7 clusters and finally, the very large group was divided into 3, 5, 7, and 8 clusters for 10, 20, 30, and 40 message respectively. Clearly, the cluster number decreases with the message size increase. This is because larger XML messages have more redundancy, which in turn causes a fewer number of clusters to be allocated to messages than is the case of smaller files. Moreover, most clusters are about equal size. This shows that a dynamic model can efficiently distribute web messages of such size, which may be more appropriate than several others.

Moreover, the average CR results achieved by applying the dynamic model to the dataset are 3.2056334, 7.2671758, 13.8358308, and 15.3180299 for small, medium, large, and very large groups, respectively, with 40 messages in the fixed-length method. On the other hand, when implementing the proposed model with the Huffman technique, the average CR results are 3.0780064, 7.8991900, 17.6439850, and 20.4135664 for small, medium, large, and very large groups, respectively, with 40 messages. Evidently, the dynamic model performance with the Huffman technique has shown better CR results than the fixed-length method. The average values obtained for CR are stated in Table 2. Table 3 presents the resultant average CR for the proposed model compared with k-means, PCA combined with k-means, and vector space clustering techniques for all groups with 40 messages and different cluster sizes. It has shown the performance of the dynamic Hilbert clustering model as better than other models (k-means, k-means+PCA) in medium, large and very large groups, as well as it has outperformed the vector space model in large and very large groups of messages.

Table 1. Clusters number results of dynamic Hilbert model for small, medium, large, and very large groups with 10, 20, 30, and 40 message

Message group	Messages number	Clusters number	Cluster size
Small	10	3	4, 2, 4
Medium		3	3, 5, 2
Large		3	3, 5, 2
Very large		3	3, 4, 3
Small	20	8	2, 2, 3, 3, 3, 2, 3, 2
Medium		6	2, 10, 2, 2, 2, 2
Large		6	4, 4, 4, 2, 3, 3
Very large		5	5, 5, 2, 5, 3
Small	30	10	4, 4, 4, 3, 2, 2, 2, 3, 4, 2
Medium		9	2, 4, 9, 2, 2, 2, 2, 5, 2
Large		6	6, 11, 5, 2, 3, 3
Very large		7	7, 3, 7, 4, 5, 5, 2
Small	40	13	4, 3, 2, 4, 5, 2, 2, 5, 2, 4, 2, 2, 3
Medium		11	2, 5, 2, 13, 2, 2, 3, 2, 2, 2, 5
Large		7	8, 14, 2, 6, 2, 4, 4
Very large		8	12, 6, 5, 2, 4, 4, 2, 5

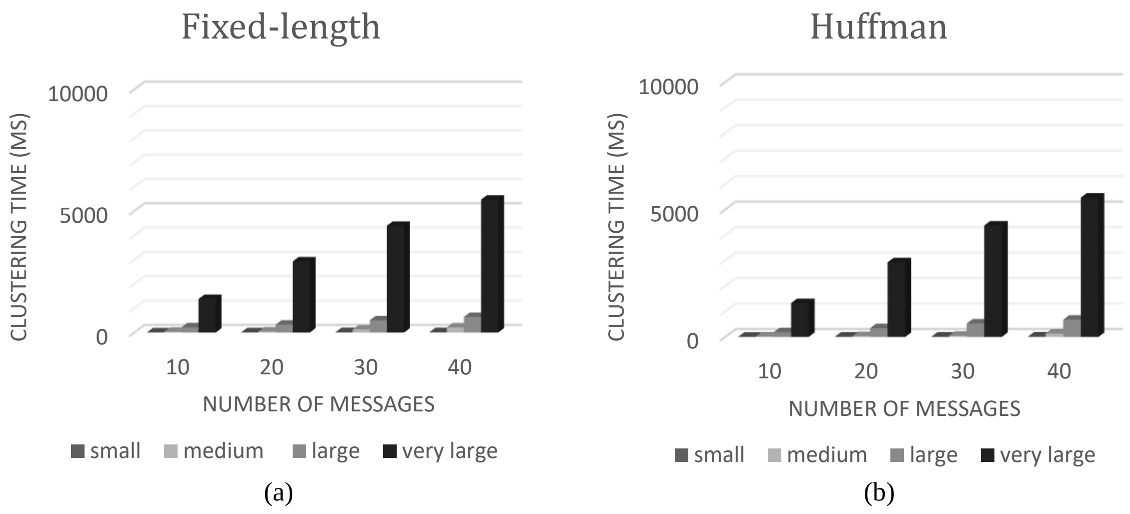
Table 2. Average compression ratio results of dynamic Hilbert model for small, medium, large, and very large groups with 10, 20, 30, and 40 message

Technique	Number of messages	Average compression ratio			
		Small	Medium	Large	Very large
Fixed-length	10	3.2697077	5.7443354	12.4924011	14.9570576
Huffman		3.1260245	5.9219548	15.3183671	19.2842577
Fixed-length	20	2.9457311	6.7048004	12.5591163	15.1736808
Huffman		2.8026315	7.2287170	15.4996723	19.7894517
Fixed-length	30	3.1320684	6.5375936	13.5885417	15.1863710
Huffman		3.0049905	6.9910348	17.3358167	20.0347384
Fixed-length	40	3.2056334	7.2671758	13.8358308	15.3180299
Huffman		3.0780064	7.8991900	17.6439850	20.4135664

Table 3. Overall average compression ratio of k-means, PCA+k-means, Hilbert, vector space for small, medium, large and very large groups with 40 messages in both fixed-length and variable-length (Huffman) compression techniques

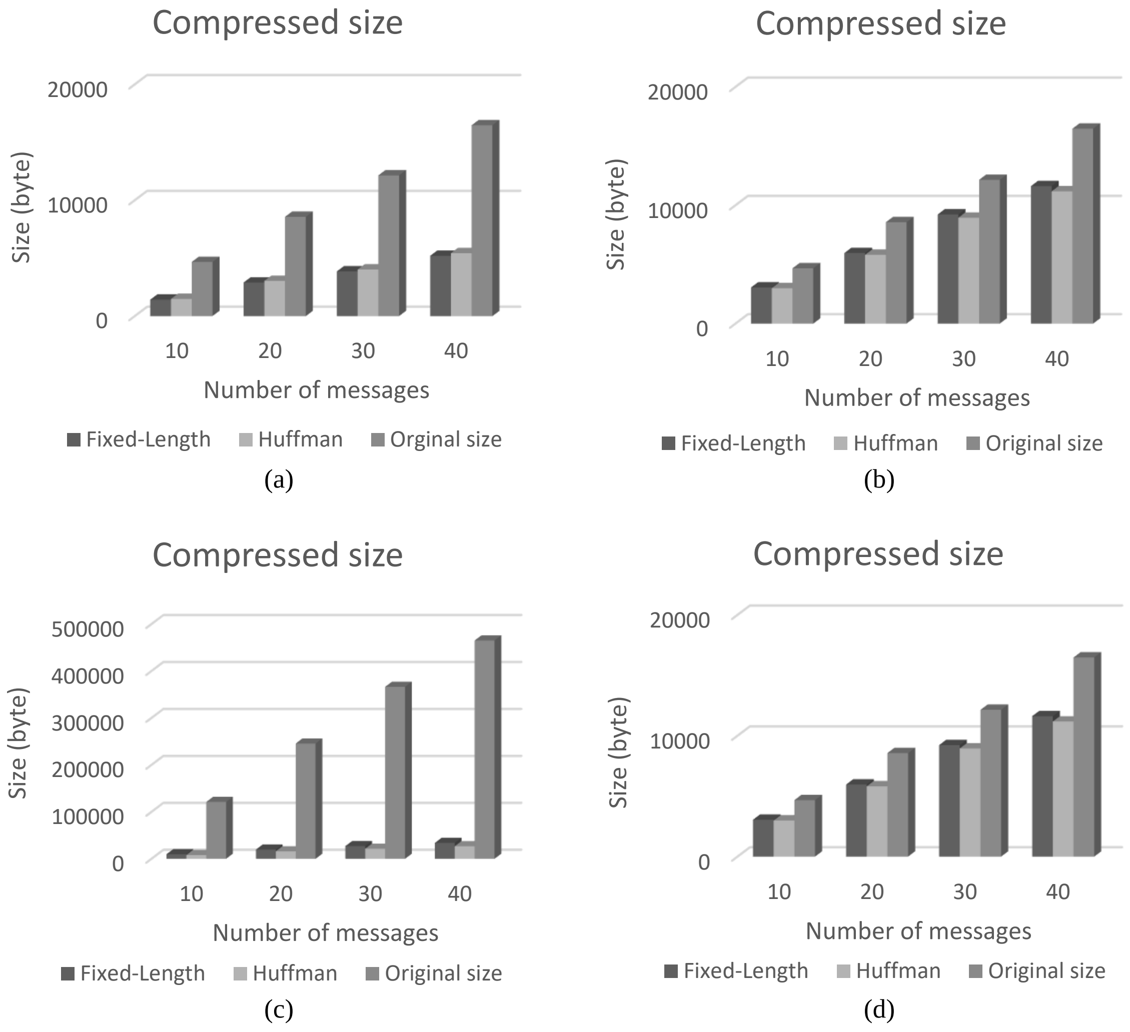
	K-means	K-means + PCA	Hilbert (dynamic)	Vector space
40 Small messages				
Fixed-length average cr.	3.920295	3.850353	3.205633	3.626756
Huffman average cr.	3.820822	3.7136433	3.078006	3.542261
40 Medium messages				
Fixed-length average cr.	6.766314	6.797503	7.267175	7.340936
Huffman average cr.	7.715699	7.843987	7.899190	7.971288
40 Large messages				
Fixed-length average cr.	12.943645	12.815021	13.835830	13.208350
Huffman average cr.	16.020012	16.279294	17.643985	16.602675
40 Very large messages				
Fixed-length average cr.	15.109293	15.127478	15.318029	15.229273
Huffman average cr.	20.163554	20.253857	20.413566	20.169220

Several illustrative figures have been computed in order to show the outcome in a clear manner. Figure 3 shows the detailed results of average clustering time for small, medium, large and very large groups with (10, 20, 30, and 40) message numbers in both fixed-length as shown in Figure 3(a) and Huffman as shown in Figure 3(b) encoding. Evidently, the clustering time has shown the best values with a small size of messages. This indicates that the needed processing time increase as the number of messages increases. Figure 4 depicts the ability to minimize the overall size of the aggregated messages with various original sizes for small as shown in Figure 4(a), medium in Figure 4(b), large in Figure 4(c), and very large in Figure 4(d). Moreover, the proposed model has achieved better results with the Huffman technique than with the fixed-length technique.



(a) (b)

Figure 3. Average clustering time of dynamic Hilbert clustering model for all groups of messages in (a) fixed-length and (b) Huffman techniques



(a) (b)

(c) (d)

Figure 4. Compressed size for (a) small, (b) medium, (c) large, and (d) very large in both fixed-length and Huffman techniques

5. CONCLUSION

In conclusion, this paper has introduced a new Hilbert clustering model based on a convex set. The main technique is computing the similarity values between web service messages and clustering them. The experimental results have shown the proposed model's effectiveness in dramatically enabling the aggregation approach to reduce network traffic compared to other traditional methods. The proposed model can achieve a high compression ratio that reaches up to 20 with the Huffman technique. Moreover, it has provided a considerable reduction in processing time. In the future, we will apply the proposed model to high dimensional dataset and use Hilbert similarity measurements for web services classification.

REFERENCES

[1] B. Natarajan, M. S. Obaidat, B. Sadoun, R. Manoharan, S. Ramachandran, and N. Velusamy, "New clustering-based semantic service selection and user preferential model," *IEEE Systems Journal*, vol. 15, no. 4, pp. 4980–4988, Dec. 2021, doi: 10.1109/JSYST.2020.3025407.

[2] S. Pawar, V. Nayak, G. Laxmi, and N. N. Chiplunkar, "A novel web service search system using WSDL," in *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*, Aug. 2020, pp. 1252–1257, doi: 10.1109/ICSSIT48917.2020.9214201.

[3] R. Elfwing, U. Paulsson, and L. Lundberg, "Performance of SOAP in web service environment compared to CORBA," in *Ninth Asia-Pacific Software Engineering Conference*, 2002, pp. 84–93, doi: 10.1109/APSEC.2002.1182978.

[4] W. J. Obidallah and B. Raahemi, "A survey on web service discovery approaches," in *Proceedings of the Second International Conference on Internet of things, Data and Cloud Computing*, Mar. 2017, pp. 1–8, doi: 10.1145/3018896.3056773.

[5] M. Kuehnhausen and V. S. Frost, "Application of the Java message service in mobile monitoring environments," *Journal of Network and Computer Applications*, vol. 34, no. 5, pp. 1707–1716, Sep. 2011, doi: 10.1016/j.jnca.2011.06.003.

[6] V. Diamadopoulou, C. Makris, Y. Panagis, and E. Sakkopoulos, "Techniques to support web service selection and consumption with QoS characteristics," *Journal of Network and Computer Applications*, vol. 31, no. 2, pp. 108–130, Apr. 2008, doi: 10.1016/j.jnca.2006.03.002.

[7] S. Subashini and V. Kavitha, "A survey on security issues in service delivery models of cloud computing," *Journal of Network and Computer Applications*, vol. 34, no. 1, pp. 1–11, Jan. 2011, doi: 10.1016/j.jnca.2010.07.006.

[8] J. Puttonen, A. Lobov, M. A. C. Soto, and J. L. M. Lastra, "Cloud computing as a facilitator for web service composition in factory automation," *Journal of Intelligent Manufacturing*, vol. 30, no. 2, pp. 687–700, 2019, doi: 10.1007/s10845-016-1277-z.

[9] X. Du, J. Xu, W. Cai, C. Zhu, and Y. Chen, "OPRC: an online personalized reputation calculation model in service-oriented computing environments," *IEEE Access*, vol. 7, pp. 87760–87768, 2019, doi: 10.1109/ACCESS.2019.2925778.

[10] K. Zeng and I. Paik, "Semantic service clustering with lightweight BERT-based service embedding using invocation sequences," *IEEE Access*, vol. 9, pp. 54298–54309, 2021, doi: 10.1109/ACCESS.2021.3069509.

[11] W. J. Obidallah and B. Raahemi, "A survey on web service discovery approaches," in *Proceedings of the Second International Conference on Internet of things, Data and Cloud Computing*, Mar. 2017, pp. 1–8, doi: 10.1145/3018896.3056773.

[12] D. Al-Shammary, I. Khalil, and Z. Tari, "A distributed aggregation and fast fractal clustering approach for SOAP traffic," *Journal of Network and Computer Applications*, vol. 41, no. 1, pp. 1–14, May 2014, doi: 10.1016/j.jnca.2013.10.001.

[13] D. Andresen, D. Sexton, K. Devaram, and V. P. Ranganath, "LYE: a high-performance caching SOAP implementation," in *International Conference on Parallel Processing*, 2004, pp. 143–150, doi: 10.1109/ICPP.2004.1327914.

[14] M. M. Hassan and H. O. Abdullahi, "Clustering simple object access protocol (SOAP) web services: a review," *International Journal of Scientific & Engineering Research*, vol. 9, no. 11, pp. 1493–1496, 2018.

[15] L. Purohit and S. Kumar, "Clustering based approach for web service selection using skyline computations," in *2019 IEEE International Conference on Web Services (ICWS)*, Jul. 2019, pp. 260–264, doi: 10.1109/ICWS.2019.00052.

[16] C. Werner and C. Buschmann, "Compressing SOAP messages by using differential encoding," in *Proceedings IEEE International Conference on Web Services*, 2004, pp. 540–547, doi: 10.1109/ICWS.2004.1314780.

[17] D. Al-Shammary, I. Khalil, Z. Tari, and A. Y. Zomaya, "Fractal self-similarity measurements based clustering technique for SOAP Web messages," *Journal of Parallel and Distributed Computing*, vol. 73, no. 5, pp. 664–676, May 2013, doi: 10.1016/j.jpdc.2013.01.005.

[18] K. A. Phan, Z. Tari, and P. Bertok, "Similarity-based SOAP multicast protocol to reduce bandwith and latency in web services," *IEEE Transactions on Services Computing*, vol. 1, no. 2, pp. 88–103, Apr. 2008, doi: 10.1109/TSC.2008.8.

[19] C. Bo *et al.*, "Development of web-telecom based hybrid services orchestration and execution middleware over convergence networks," *Journal of Network and Computer Applications*, vol. 33, no. 5, pp. 620–630, Sep. 2010, doi: 10.1016/j.jnca.2010.03.025.

[20] D. Al-Shammary and I. Khalil, "Dynamic fractal clustering technique for SOAP web messages," in *2011 IEEE International Conference on Services Computing*, Jul. 2011, pp. 96–103, doi: 10.1109/SCC.2011.15.

[21] N. G. Rezk, A. Sarhan, and A. Algergawy, "Clustering of XML documents based on structure and aggregated content," in *2016 11th International Conference on Computer Engineering and Systems (ICCES)*, Dec. 2016, pp. 93–102, doi: 10.1109/ICCES.2016.7821981.

[22] 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*, Aug. 2010, pp. 6174–6177, doi: 10.1109/IEMBS.2010.5627759.

[23] J. H. Hwang and M. S. Gu, "Clustering XML documents based on the weight of frequent structures," in *2007 International Conference on Convergence Information Technology (ICCIT 2007)*, Nov. 2007, pp. 845–849, doi: 10.1109/ICCIT.2007.101.

[24] G. Yongming, C. Dehua, and L. Jiajin, "Clustering XML documents by combining content and structure," in *2008 International Symposium on Information Science and Engineering*, Dec. 2008, vol. 1, pp. 583–587, doi: 10.1109/ISISE.2008.301.

[25] "Web standards," *W3C*, https://www.w3.org/standards/ (accessed Jun. 18, 2023).

[26] H. Liu, H. Bao, J. Wang, and D. Xu, "A novel vector space model for tree based concept similarity measurement," in *2010 2nd IEEE International Conference on Information Management and Engineering*, 2010, vol. 2, pp. 144–148, doi: 10.1109/ICIME.2010.5477749.

[27] Jianghui Liu, J. T. L. Wang, W. Hsu, and K. G. Herbert, "XML clustering by principal component analysis," in *16th IEEE International Conference on Tools with Artificial Intelligence*, 2004, pp. 658–662, doi: 10.1109/ICTAI.2004.122.

BIOGRAPHIES OF AUTHORS

Nawras A. Al-Musawi received her M.Sc. degree in computer science from the college of computer science and information technology at the University of Al-Qadisiyah-Iraq in 2022. She has obtained a Bachelor degree in computer science from the College of Science at the University of Al-Qadisiyah-Iraq in 2009. She has worked in teaching of computer science courses for high school students for several years. Nawras has an interest in research on web services, data compression, clustering techniques and encoding methods. She can be contacted at email: nawras.kadhim@qu.edu.iq.

Dhiah Al-Shammary has received his Ph.D. in Computer Science in 2014 from RMIT University, Melbourne, Australia. Dhiah is awarded as the best Ph.D. student and top publication during his Ph.D. period. He has several years experience in both education and industry. His main industrial experience came from Silicon Valley based companies working on security projects including non-traditional and quantum scale encryption. Dhiah has worked at several universities in both Australia and Iraq like RMIT University and University of Al-Qadisiyah. His research interests include performance modelling, web services, compression and encoding techniques, and distributed systems. Dhiah has several publications in the areas of improving the performance of web services and encoding techniques. He can be contacted at email: d.alshammary@qu.edu.iq.