INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

Dynamic Compilation of Pattern based clustering and Volumetric Probabilistic Mining for Network Routing in Cognitive Radio Sensor Networks

Kumaresh Sheelavant[1]*, R Sumathi[2]

1 Research Scholar, Dept of CSE, Siddaganga Institute of Technology, Tumakuru, 572103, Karnataka, India
2 Professor, Dept of CSE, Siddaganga Institute of Technology, Tumakuru, 572103, Karnataka

OPEN ACCESS

Received: 03.11.2021
Accepted: 20.11.2021
Published: 06.12.2021

Citation: Sheelavant K, Sumathi R (2021) Dynamic Compilation of Pattern based clustering and Volumetric Probabilistic Mining for Network Routing in Cognitive Radio Sensor Networks. Indian Journal of Science and Technology 14(41): 3093-3106. https://doi.org/10.17485/IJST/v14i41.1838

* **Corresponding author**.

kums999@gmail.com

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Objectives: The key objective is to investigate the current state of the art in web service prediction systems and also to improve the retrieving process with improved accuracy and to reduce the searching time**.** As well as to enhance the performance of data validation, quality of services and speed of the process. **Method:** In this study, an advanced model of the Dynamic Compilation of Pattern (DCP) method with a Lexical Subgroup (LS) system was used to estimate the similarity between the request data and the entire network. These are all indexed and grouped as a cluster to form a paged format of network structure which can reduce the computation time during the searching period. Also, with the help of prediction, the relevancy of feature attributes in the network is predicted, and the matching index is sorted to provide the recommended data for given request data. This was achieved by using Volumetric Probabilistic Mining (VPM). **Findings:** The performance of the proposed DCP-VPM is proved through extensive simulations and compared to those of the state-of-the-art methods. On the average, it is realized that the DCP-VPM always outperforms EACRP, ERP, ESUCR and ESAC related to minimizing average energy consumption, packet delivery ratio, end-to-end delay at different number of clusters by 10.2%, 18.6%, 11.3%, 12.5% compared to EACRP, ERP, ESUCR and ESAC respectively. Proposed cluster-based routing technique out performs all other routing techniques. **Novelty:** Route based request prediction system was focused to predict and analyse data from the network. That is why enhanced clustering, distance-based similarity and retrieving mechanism are used. Irrelevant parameters removal and ordering are performed using DCP with LS system. Then nodes are processed for learning model using VPM prediction model. Finally, as the recommended result for the routing application, the matched data that is related to the request input is listed.

Keywords: Dynamic Compilation of Pattern; Lexical Subgroup system; Volumetric Probabilistic Mining; Request prediction; Cognitive Radio

1 Introduction

The Cognitive Radio Sensor Network (CRSN) plays an important role in the routing system's optimal network selection process by performing matching prediction and similarity identification. In the routing system, this type of data searching becomes complex in searching for the best match for given request data and fails in the accuracy of the request prediction process. It enables the nodes to rent out the spaces on their physical machines with increased profit maximization. The network routing environment is classified into homogeneous and heterogeneous architectures. In a homogeneous architecture, the entire service is offered by a single vendor, and in a heterogeneous architecture, the service contains components integrated by various vendors. In the request prediction process, the routing concept was majorly focused on presenting the sorted list of subjects and courses. information by referring to the network. Traffic in Routing focused on analysing the data clusters with labelled properties that can be characterised based on the ratings, probability of visit, and other parameters. A Web service is a platform-independent factor that is mainly used to help with machine-to-machine communication in a network[1]. In this, the quality of service is one of the major considerations for users when selecting their required services, which describes the non-functional characteristics of the web services. Normally, QoS[2] is defined as the set of characteristics of data availability, reputation, and throughput. Normally, the processes of service selection and prediction are the major things that are used to enable the service composition in recent years[3].

The traditional work developed a similarity-based request prediction system. It does not satisfy the user's requirements by providing the most similar services[4]. Moreover, it utilized some prediction and clustering mechanisms during service storage and routing. The existing network clustering techniques such as K-means[5], K-Medoids[6], and Fuzzy C-Means (FCM)[7,8] are not highly efficient for request prediction, because they have the major issues of being highly sensitive, requiring a finite number of clusters, and large searching space. The major reasons for using the prediction[9] techniques are to ensure the confidentiality, privacy, integrity, access control, and authentication of the data. So, these techniques are also not highly suitable for accurate request prediction[10]. Figure 1 shows the basic block diagram of the prediction system in the routing process.

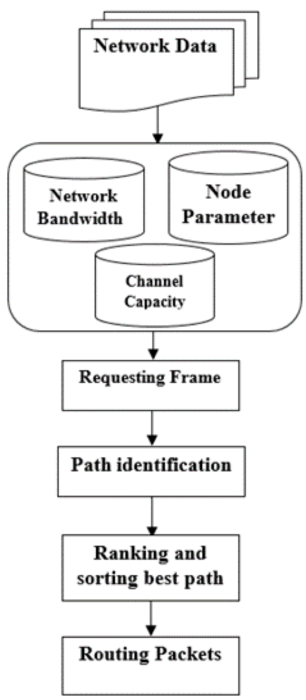

Fig 1. Generic block diagram of prediction system in network routing

Here in the above diagram by using web sources prediction system extracts the user, feedback and dynamic data by applying query processing and knowledge engine, user query will be applied to this. Finally, user will get the recommended results.

Liu, et al[11] developed a location aware personalized collaborative filtering mechanism for improving the QoS of the prediction system. In this mechanism, both the location of the users and the web services were leveraged for electing the target service or the user. The main QoS factors that were considered in this work were availability, response time, user dependency, and reliability.

Here, the similarity computation was performed by selecting the similar neighbors with the use of the weighted PCC technique. Also, the impact of sparseness was examined in this work for evaluating the accuracy of prediction. Schnabel, et al [12] analyzed the shortlists for supporting the user decision process in a prediction system. In this paper, it was stated that the short lists provided a better improvement in both downstream and user satisfaction performance. Also, it improved the quality of prediction by implementing additional feedback. Xu, et al [13] suggested a context aware QoS prediction scheme for the user prediction system. Here, the mapping relationship between the geographical distance and the similarity value was analyzed on the user side.

Yao, et al [14] developed a new approach by integrating collaborative filtering with the content-based filtering technique for service prediction. In this system, the semantic content and rating data were utilized to perform the prediction through the use of a probabilistic generative model. Moreover, three main requirements such as prediction serendipity, recommending newly deployed services, and prediction accuracy were examined in this work for developing an efficient prediction system. Then, a three-way aspect model was implemented to identify the similarities of the users based on the semantic contents of web services. Kang, et al [15] suggested three different prediction approaches, such as the collaborative filtering approach, the content-based approach, and the hybrid approach, for developing an efficient service prediction system

To consider the request prediction system for the routing network, Bouihi, et al [16] proposed a CRSN based context prediction system and mobile routing applications. In this, the context types like profile, social interactions, learning activities, and device specifications were considered from the network in the learning object CRSN. The OWL rules are used for the filtering of context to recommend the request input. In Liu et al [17], paper presented a survey of different methodologies for the prediction system based on the CRSN in routing. In that, it states that the hybridization of algorithms and other prediction techniques achieved better similarity identification models based on the knowledge-based prediction system. Later, in Zareei et al [18], the authors proposed a course prediction system based on the request classification approach. This estimates the relevant data for the request input by using the classification of request data from the network.

The CRSN calculates the N-List of relevant network features that match the request input and displays the recommended course information. Similarly, in George et al [19], the paper presented a review of the CRSN based routing process. The recommender systems specified in the analysis use CRSN, artificial intelligence, among other techniques, to provide personalized predictions. This helps to prepare the learning libraries and feature routing models to enhance the prediction system in the routing process. In Prajapat et al [20], author proposed a novel learning path prediction model. This was based on the multidimensional knowledge graph framework for the routing system. This multidimensional knowledge graph method was used to separate the overall network and organized into several classes. This will enhance the learning capacity and reduce the time complexity of the classification model. Similarly, in Bala Vishnu et al [21], paper proposed a novel prediction system for the routing process using the Moodle Routing platform. This identifies the similarity of course information from the network and retrieves the relevant data. MoodleRec performs the sorting of supported standard compliant learning object repositories and suggests a ranked list of learning objects that are similar to the request input that are operated at two different levels of classification.

From the above findings, it is investigated that the existing approaches have the following drawbacks:

- It failed to recognize the QoS variations.
- The existing systems require more learning set to retrieve for request data.
- This also increased time complexity of the memory-based prediction systems.
- It offered a list of ranked services with no transparency.

To solve these problems, this paper aims to develop a new request prediction system. In this, the request searching and relevant data identification was processed by similarity indexing and the DCP based cluster estimation. The VPM based classification algorithm enhanced the accuracy performance of network routing system than by using the traditional prediction model. The detailed description about the proposed work is explained in section II

2 Proposed Work

In this section, the detailed description of the proposed methodology is presented with its flow illustration. The intention of this paper is to perform request prediction in an efficient manner. In this system, the progressive data like student educational course data is given as the input, which is pre-processed by performing the irrelevant parameters removal and ordering.

After getting the filtered / pre-processed data, the matrix is generated for selecting the Cluster Pattern (CP) based on the normalized DCP method. Based on the CP, the clusters are formed by implementing the LS system, and the nodes in each cluster are then processed for the learning of classification algorithm by using VPM. After that, the similarities such as Kolomogorov and Transformation distance are computed to identify the similar items. Consequently, the n numbers of attributes of the nodes are stored in the architecture server. When the server receives the request from the user, it retrieves the relevant data by searching for the best match for the request data and recommending the result by using the VPM classification model. Based on the highest similarity value, the services are recommended to the requested user. Finally, as the recommended result for the routing application, the matched data that is related to the request input is listed.

Figure 2 shows the block diagram of the proposed routing prediction work. In this, the input data is pre-processed and arranged with the attributes. This will form the cluster of feature vectors that extract the similarity between the attributes of the feature set. Then from that feature arrangement, the dataset was partitioned for learning and network data. The learning data was passed to the feature learning of classification, which forms the neurons present in the prediction model. Then, according to the match prediction in the dataset, the network data is classified and the relevant data is sorted for the prediction process. This was processed by using the proposed VPM prediction model. Then they enhanced the routing prediction model compared to the traditional method of the classification process.

The stages that involved in this system are as follows:

- Preprocessing
- Dynamic Compilation of Pattern
- Clustering of nodes
- Similarity Estimation
- Service Routing and Prediction

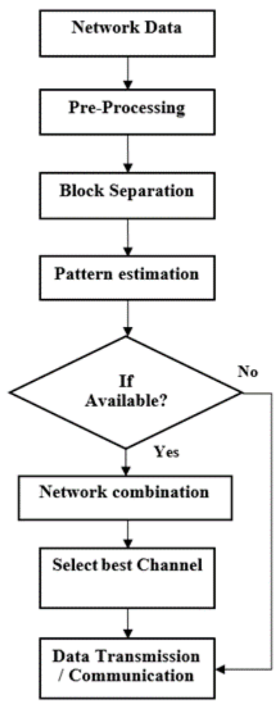

Fig 2. Overall block diagram of proposed work

2.1 Pre-processing

First, the dataset is given as the input for preprocessing, where the irrelevant parameters are removed and ordering is performed to obtain the filtered data. The main intention of dataset preprocessing is to optimize the data size by selecting attributes that are related to the prediction process. Here, the unwanted characters or letters are filtered to reallocate the special characters that are needed in the dynamic analysis model. The special characters are considered as the Unicode value to represent the letter size, which can reduce the data memory size. This improves the prediction quality with better routing accuracy. The filtering of irrelevant data can be identified by estimating the uniqueness of the attribute value, if that value can be segmented by its related components.

2.2 Dynamic Compilation of Pattern based pattern generation

After pre-processing, the matrix is generated for the pre-processed data by using the DCP. It is also known as the universal distance measure that finds the distance between each object. Moreover, it simultaneously uncovers all similarities for selecting the CP. In this stage, the set of nodes and its domain list are given as the input. Here, each domain contains a set N file, which are stored in a repository R_N and its size is denoted as M. To construct the matrix, the varying number of parameters in each node are extracted i.e., K_i and K_j. Then, the bytes of K_i and K_j are also extracted and stored in a separate variable wd_x, wd_y and wd_{xy}. Consequently, the binary values are calculated for the extracted bytes of data, based on this the values of M_{xy}, *and* N_{xy} are computed as shown in below:

$$M_{xy} = \begin{cases} |B_x * B_x| & \text{if } (B_x > B_y) \\ |B_y * B_y| & \text{Otherwise} \end{cases} \tag{1}$$

$$N_{xy} = \begin{cases} N_{xy} & \text{if } (N_{xy} > N_{yx}) \\ N_{yx} & \text{Otherwise} \end{cases} \tag{2}$$

Then, the distance $dist_{xy}$ is computed by the ratio of $\frac{1}{log(Derv_{xy})}$ and dif_v. Consequently, the T_{dist} is updated with the sum of T_{dist} and $dist_{xy}$.

$$dist_{xy} = 1 - \frac{(2 * (\frac{1}{log(Derv_{xy})} - min_v)}{dif_v} \tag{3}$$

$$T_{dist} \leftarrow Update\ (T_{dist} + dist_{xy}) \tag{4}$$

Then, the value of O_{dist} is updated with the values of T_{dist}, and finally, the M_{NID}(i, j) is estimated based on the updated O_{dist}/M.

Algorithm I - Dynamic Compilation of Pattern (DCP) Algorithm
Input: Set of nodes, Domain list
Output: DCP matrix (M_{NID})
Step 1: Let, N be the set of Domains, in which each domain contains a sample set of 50 files;
Step 2: Let, R_N be of the repository of files, which holds the set of N nodes;
Step 3: Let, M be the Total Size of R_N;
Step 4: To construct the M_{NID} of size (M, M)
Step 5: For i=1 to M
$K_i \leftarrow$ extract Parameters (R_N(i))
For j=1 to M
$K_j \leftarrow$ extract Parameters (R_N(j))
For x=1 to size (K_i)
For y=1 to size (K_j)
wd_x= extract bytes ($K_i(x)$)
wd_y= extract bytes ($K_i(y)$)
wd_{xy}= extract bytes ($K_i(x) * K_i(y)$)
B_x= compute fold (wd_x)
B_y= compute fold (wd_y)
B_{xy}= compute fold (wd_{xy})
Compute M_{xy} by using equation (1)
Compute N_{xy}, N_{yx} by using equation (2)
N_{xy}= dif (B_{xy}, B_x)
N_{yx}= dif (B_{xy}, B_y)

If ($N_{xy} > N_{yx}$)
NK_{xy}= N_{xy}
Else
Nk_{xy}= N_{yx};
End if;
$Derv_{xy}$= M_{xy}/ NK_{xy}
The distance $dist_{xy}$ and T_{dist} are computed by using equation (3) and (4);
End for x
$O_{dist} \leftarrow$ Update ($O_{dist} + T_{dist}$)
End for y
M_{NID}(i, j) $\leftarrow$ Update (O_{dist}/M);
End for i
End for j

2.3 Clustering of nodes

After selecting the CP using DCP, the number of clusters are formed by implementing the LS technique. When compared to traditional clustering techniques such as k-means, and fuzzy c-means, it is an efficient clustering technique widely used in the field of computer science. Because, it provides a high quality of clusters by iteratively exchanging messages between all pairs of data. The major reasons for using this algorithm are reduced clustering errors, deterministic, increased efficiency, and simple computation. Also, it does not require satisfying the triangle inequality, because it supports the similarities. Moreover, the major characteristics of this technique are availability and responsibility. This algorithm computes a set of exemplars for representing the dataset, where the pair-wise similarity is estimated between each pair of data. Here, the sum of distances or similarities between all the data points with respect to their equivalent exemplars is maximal. In this algorithm, the availability matrix S_i and the responsibility matrix S_j are constructed based on the distance matrix M_{NID} obtained from the previous stage. Then, these matrices are updated by checking the rows and columns in M_{NID} is greater than the value of A_{ij}.

$$A_{ij} = \begin{cases} 0.5 & \text{if } \left(M_{\text{NID}}\,(i,j) <= 0.5\right) \\ 0 & \text{Otherwise} \end{cases} \tag{5}$$

Consequently, the exponential matrix is constructed by checking the value of sum of A_{ij} and R_{ij} is greater than 0.

$$\mathrm{R_{ij}} = \begin{cases} \mathrm{M_{NID}(i,j) - A_{ij}} & \text{If } \left(\mathrm{A_{ij}} <= \mathrm{M_{NID}(i,j)}\right) \\ 0 & \text{Otherwise} \end{cases} \tag{6}$$

Then, the average for the Max $(Expm_i)$ and $R_{i(Idx_x)}$ is computed and updated with the avg_{list}.

$$\mathrm{Expm_{ij}} = \begin{cases} 1 & \text{if } \left(A_{\mathrm{ij}} + \mathrm{R_{ij}}\right) > 0 \\ 0 & \text{Otherwise} \end{cases} \tag{7}$$

$$avg \ = \sum_{x=1}^{size(Idx_{ls})} R_{i(Idx_x)} \tag{8}$$

Where, i and j are the size of the matrix $(S_i,\ S_j)$, $Idx_{ls} \leftarrow$ Max $(Expm_i)$ where i is the size of Matrix, and S_i - is the Index list with maximum elements of each column in the matrix. After that, the distance index list is estimated by finding the difference between the $avg^2$and $avg_x{}^2$. Then, the maximum index of the dis_{ls} is calculated, based on this value, the CP is selected for each cluster.

$$avg_R = \frac{\sum_{j=1}^{S_j} R_{ij}}{S_j} \tag{9}$$

$$dis_{ls} = \sqrt{avg^2 - (avg_x{}^2)} \tag{10}$$

Where, x is the Size of matrix S_j, and C_{id}= Max (Index (dis_{ls})).

Algorithm II –Clustering
Input: Distance Matrix [DCP matrix (M_{NID})]
Output: clusters,A_{ij},R_{ij}
Step 1: Construct Availability Matrix and responsibility matrix
Let S_i and S_j be size of matrix (M_{NID}) and Set K=2;
For i=1 to S_i
For j= 1 to S_j
$A_{ij}$is computed by using equation (5)
End for j;
End for I;
Step 2: Construct and update responsibility matrix and Availability matrix;
For X_k= 1 to k
For i=1 to S_i
For j= 1 to S_j
$R_{ij}$is computed by using equation (6);
End for j
End for i
For i=1 to S_i
For j= 1 to S_j
Let $temp\_R_{ij}$=0;
For m=1 to S_i
$temp\_R_{ij}$=$temp\_R_{ij}$+R_{im}
End for m;

$$temp\_R_{ij} = \begin{cases} temp\_R_{ij} + R_{ij}\ if(temp\_R_{ij} \leq 0) \\ R_{ij} \quad Otherwise \end{cases}$$

If $(i! = j)$

$$A_{ij} = \begin{cases} 0 & if(temp\_R_{ij} \leq 0) \\ temp\_R_{ij} & Otherwise \end{cases}$$

Else

$$A_{ij} = \begin{cases} temp\_R_{ij} + R_{ij}\ if(temp\_R_{ij} > 0) \\ R_{ij} \quad Otherwise \end{cases}$$

End if
End for S_j
End for S_i
End for X_k
Step 3: Compute Exponential Matrix by using equation (7);
Update avg_{list}←avg;
For y= 1 to S_j
Compute avg_R by using equation (9);
Compute the distance list by using equation (10);
Update $C_{id} \rightarrow C_{head}$
End for y;

2.4 Similarity Computation

In this stage, the similarity is computed between the nodes by computing the Kolmogorov and transformation distance-based similarity measures. In which, the kolmogorov is a widely used similarity mechanism that encodes a finite set of objects into strings denoted as {0, 1}. For instance, it estimates the similarity between two representations (i.e. A and B), where it takes A as input and B as output. Then, the quantity of this similarity is denoted as the K(B|A), which is semi-computable. Then, the transformation distance-based similarity measure is a kind of asymmetric technique and it does not have admissible distance.

In this technique, the user request is given as the input, and the estimated similarity is resulted as the output. Here, the size of cluster head nodes and the parameters in each cluster head node are computed, then the size of the node is converted into bytes. Also, the binary folds are computed for the bytes of data, from those the minimum and maximum folds are estimated by using the following equation:

$$\begin{cases} \begin{cases} (\mathrm{B_{fx}} * \mathrm{B_{fx}}) = \mathrm{Max}\,(\mathrm{B_{xy}}) \\ (\mathrm{B_{fy}} * \mathrm{B_{fy}}) = \mathrm{Min}\,(\mathrm{B_{xy}}) \end{cases} & \text{if } (\mathrm{Bf_x} > \mathrm{Bf_y}) \\ \begin{cases} (\mathrm{B_{fy}} * \mathrm{B_{fy}}) = \mathrm{Max}\,(\mathrm{B_{xy}}) \\ (\mathrm{B_{fx}} * \mathrm{B_{fx}}) = \mathrm{Min}\,(\mathrm{B_{xy}}) \end{cases} & \text{Otherwise} \end{cases} \tag{11}$$

$$p1 = \left(\frac{(Bf_{xy} - Min\ (B_{xy}))}{Max\ (B_{xy})} \right) \tag{12}$$

$$p2 = \frac{((Size\ (S1) - Size\ (S2))}{size\ (S12)} \tag{13}$$

Based on these values, the transmission distance similarity is estimated by the product of p1 * p2. Then, the KC is computed by generating the mask value for the binary folds of the data.

$$Sim_{Kc} = \left(\frac{(K_{xy} \&\& Mask) - Min\ (B_{xy})}{Max\ (B_{xy}) - Min\ (B_{xy})} \right) \tag{14}$$

At last, the similarity values of both TCD and KC are summed for estimating the total similarity. It is used to identify the most similar items related to the user request.

Algorithm III — Similarity Computation
Input: User Request
Output: Estimated similarity
Step 1: Let U_Q be the User Request
Step 2: Let K_U be the Parameters in the User Request
Step 3: The Server ID K_U with the cluster Keys.
Step 4: Let c_{Head} be the cluster head nodes and kc_H be the parameters in the cluster head nodes
For M =1 to Size of (K_U)
For N = 1 to Size of (kc_H)
S1=$K_U(M)$, S2= $kc_H(N)$ and S12= s1 * s2
B1, B2 and B12 bytes for of S1, S2 and S12
Bf_x, Bf_y and Bf_{xy} be the binary folds of B1, B2 and B12;
Compute Max (B_{xy}) and Min (B_{xy}) by using equation (11);
Compute p1 and p2 by using equation (12) and (13);
Sim_{TCD}= p1* p2;
K_{xy}=$Bf_{xy} - Min\ (B_{xy})$
Set Mask=0xFF;
Compute Sim_{Kc} by using equation (14)
Sim_{Tot}=(Sim_{TCD}+Sim_{Kc})
End for N
$C_{Ch} = Min$(Index (Sim_{Tot}/N));
End for M

2.5 Request Prediction

Finally, the routing is provided for the items based on its similarity values, where the items that have high similarity values are recommended to the requested users. Here, VPM is used to rank the items based on their similarity. In this technique, the user request and selected CP are given as the input, and the matched node for the request is provided as the output.

At first, the server retrieves the parameters in the user request K_U. Then, the parameters in the number of nodes K_{Dn} are extracted. After that, the similarity between the user request and the parameters in the node is computed, based on this the list of similarity score Sim_{Ku} is estimated for the number of nodes in the matched cluster. Finally, the matched node E_{rank} is listed and display as the recommended information about the request data for classification process.

The Figure 3 shows the architecture of proposed VPM prediction model with the prediction technology. In that, the input sequences are formed as the cluster of data based on the similarity identification between each attribute of the input data. Then from that cluster matrix, the neurons are formed from $N_\alpha^1(x)$ to $N_\alpha^M(x)$ based on its weight value. From that, the correlation between these network data is extracted and make it as the per the matching rank. Then by estimating the normalization of overall data, these are sorted and arranged according to the prediction match. This was performed by using the batch normalization method that estimate the normalization between attributes of feature set. The Logistic regression model evaluates the best matching for the request data and retrieve the label of it. The algorithm steps of the proposed VPM prediction model are described in the algorithm IV.

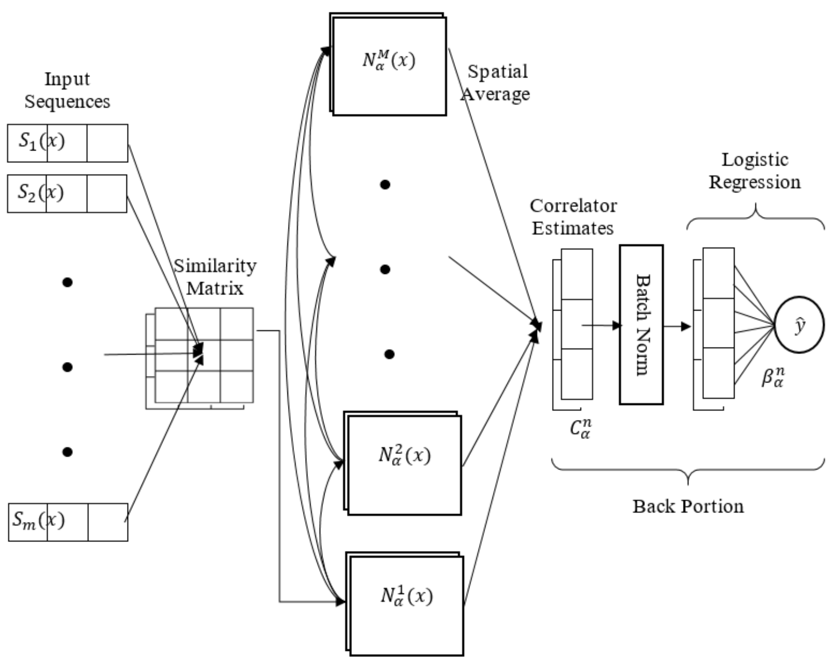

Fig 3. Architecture diagram of proposed VPM

Algorithm IV – VPM
Input: User request, Chosen cluster head C_{Ch}
Output: Matched node for the Request
Step 1: Let U_Q be the User Request
Step 2: Let K_U be the Parameters in the User Request
Step 3: The Server index K_U with the cluster Keys.
Step 4: Let N be the set of nodes in the cluster;
Step 5: For I =1 to N (No of nodes)
K_{Dn}= Parameters in the Node
$Sim_{Ku} \leftarrow$ Similairity (K_{Dn},K_U) and update
End for N
Step 6: Sim_{Ku} is the list of similarity score for the nodes in the matched cluster.
$E_{rank} = \frac{Min(Sim_{Ku})}{\sum_{m=1}^{N} Sim_{Ku}}$
Matched node is denoted as N (E_{rank})
Step 7: Decrypt the matched node based on the above mentioned non-abelian algorithm;

3 Result Analysis

The proposed model is compared with existing methods [22], DCP-VPM outperforms some traditional methods in-terms of packet delivery ratio, end-to-end delay and throughput. The routines for the method and iterations were created with in Python simulation software for this phase of the research, the existing techniques takes 100 iterations as compared to the DCM-VPM which takes 50 iterations to meet all QoS requirements, which is simulated and conducted using a widely available network architecture. Figure 4 depicts a CRSN network structure with a random creation of 20 nodes to 38 links. To demonstrate the network performance of proposed technique, it is compared with existing works presented in [22]. In all the simulations, nodes are deployed randomly in the area of interest with field dimension of 250 m $\times$ 250 m in a plane area. In the CRSN architecture, the blue circle symbolizes the network nodes, while the line that connects the nodes represents the fibre cables.

Network parameters and starting traffic level values with in routing path was initialized and validated using that data. This section also compares the efficiency of the suggested routing optimization model to other current routing models by computing comparative parameters to indicate the efficiency of the DCP-VPM method.

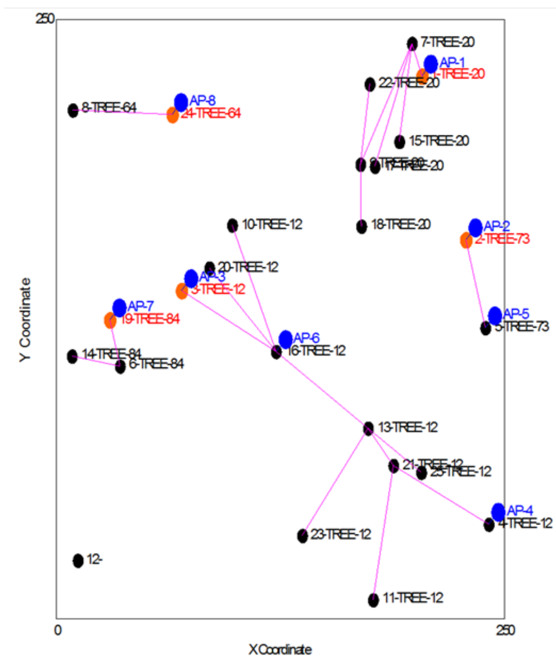

Fig 4. Network Architecture of CRSN

For a network design of 20 nodes and 38 links based random network architecture, the proposed work of optimal routing system may be compared and justified with current systems of EACRP, ESAC, and Adaptive Clustering in [22]. This can be justified because the suggested algorithm can address the issue in dynamic network design because the architecture used for the comparison is arbitrary formation. This was justified because the suggested algorithm can address the issue in dynamic network design because the architecture used for the comparison is arbitrary formation.

Packet Delivery Ratio - PDR (%), Throughput (Bits/s), Energy consumption and time delay are the parameters used to validate the suggested approach (J). The PDR (%) of a method can be represented as the ratio between the number of rejected connections by the nodes and the total number of available connections in a network. Because the rejected connection can be estimated from the connections that are timed out of communication due to pattern unavailability or by the routing time based on traffic in a network path.

From this network arrangement, the PDR for the proposed work compared with the existing methods of EACRP, ESAC and Adaptive Clustering is shown in the Figure 5 separated by colour code for each method. In that, the x-axis of the graph representing the number of nodes in the architecture in the rage of 0 to 20 numbers the y-axis of the graph represents the packet delivery ratio in (%). From the graph, it shows the trend line of PDR (%) that represents that the proposed DCP-VPM algorithm achieved better results in increasing the PDR value compared to other existing methods of routing.

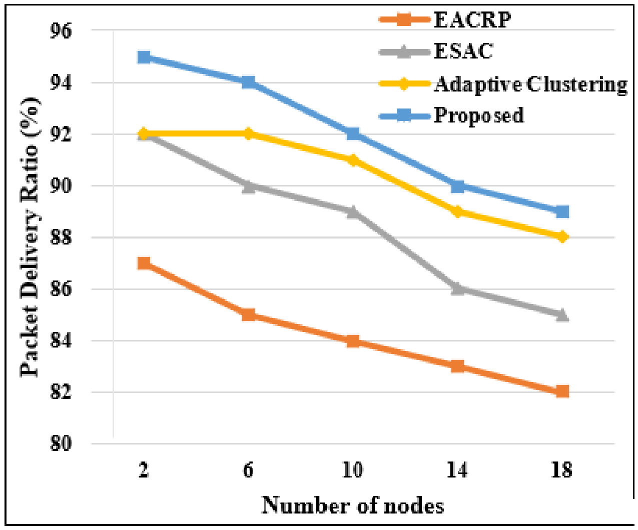

Fig 5. Comparison results of Packet Delivery ratio (%)

In Figure 6, To check the performance of energy consumption of the proposed routing method with respect to number of nodes in the network architecture. average energy consumption is evaluated against different number of nodes. It shows that proposed routing protocol gives improved result over the ESUCR, EACRP and ERP due to effective transmission mechanism. In intra cluster data transmission, the node head [22] assigns slots to cluster members based on channel residual time. It reduces the packet collision and number of retransmission of packets. As the node density increases the average energy consumption increases due to reduced opportunities for SUs transmission.

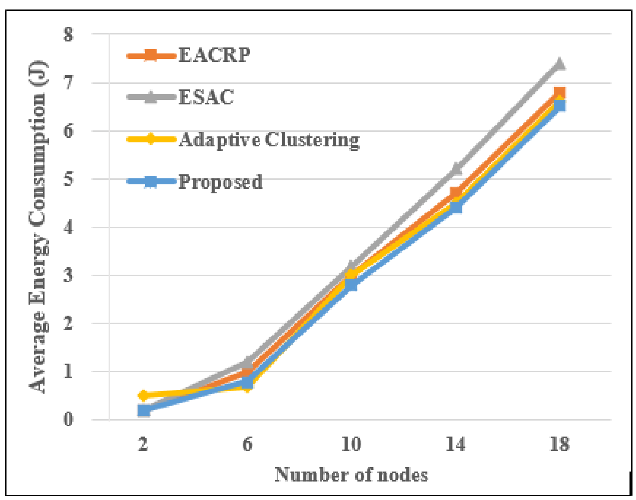

Fig 6. Comparison of Average Energy Consumption (J)

To evaluate the delay performance or time consumption of proposed routing method, this can be compared with the paper [23]. The Figure 7 shows the comparison result of average end-end delay time in (sec) for different number of nodes shows that average end-to-end delay is low in low Primary User (PU) density. It increases as the PU density increases since channel occupancy decreases for Secondary Users (SU) at high PU density. The average end-to-end delay of the proposed scheme is better than ERP, ESAC since proposed scheme sends the data from source node to sink node by selecting most energy efficient cluster with highest channel residual time. It reduces the collision probabilities with PU and packet drops which improve the average end-to-end delay.

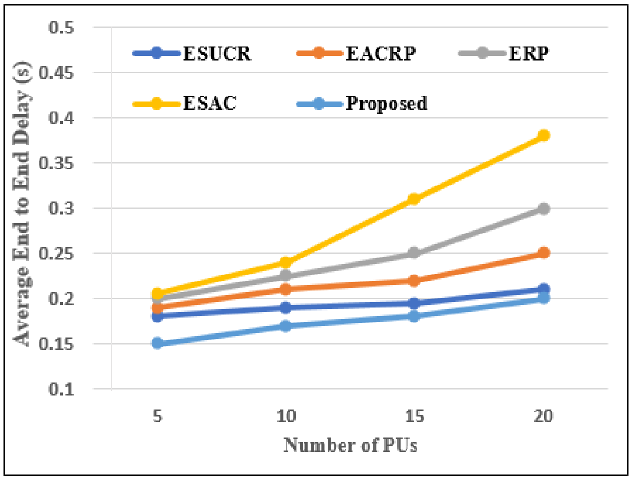

Fig 7. Average End to End Delay comparison

From this the packet loss for each time interval (sec) was estimated and presented with the comparison of existing system. The Figure 8 shows the comparison chart of amount of packet loss (Bytes) for increase in time interval. The proposed DCP-VPM out-performs all other routing techniques. Also, this was tested in the network architecture of[(24,25)] based CRSN arrangement.

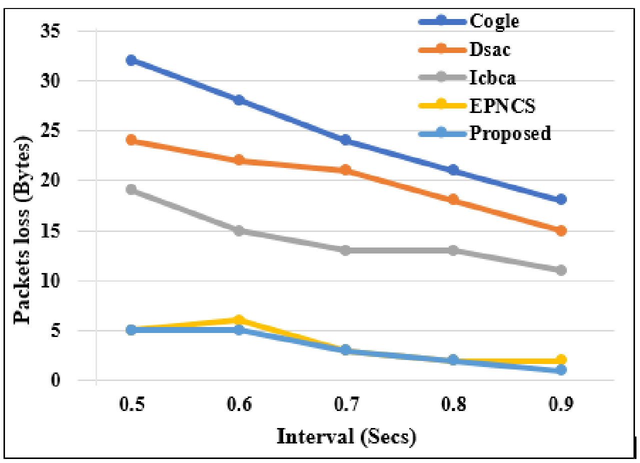

Fig 8. Packet loss at every time interval

Figures 9 and 10 represents the plot for analysing the parameters of throughput (bits/s) and PDR (%) respectively. The suggested DCP-VPM technique may handle the routing problem in a CRSN with optimal packet transmission with in routing path, depicts that packet delivery ratio decreases as number of events increases. This happens due to networks have to hold more data transmissions as number of event increases. It results in collisions and interference with further data packets generated from the events. However, rate of decrease in proposed routing protocol is low as compared with Cogle, Dsac, Icbca and EPNCS. The channel residual time and optimum transmission distance reduces the PU interference and distribution of the load, respectively. according to the results of this test study with various settings.

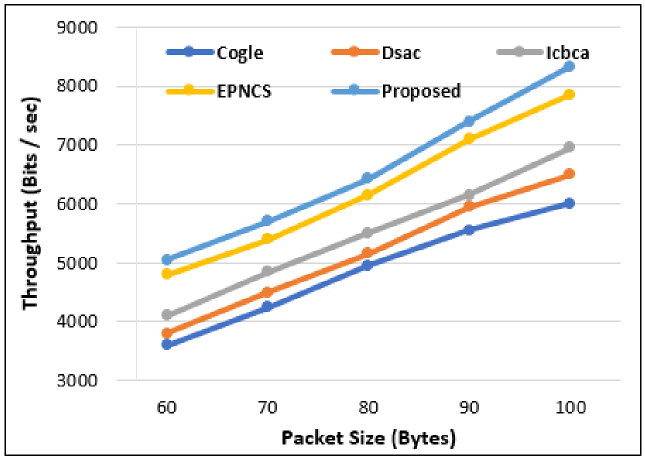

Fig 9. Packet Size (Bytes)vs Throughput (Bits / s)

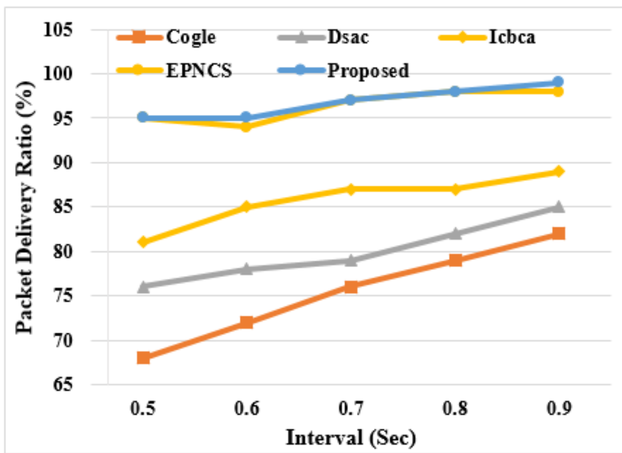

Fig 10. Packet Delivery Ratio (%) for each interval of time

4 Conclusion and Future enhancement

End users lacks reliable services, in order to overcome these difficulties; this study designed a new request prediction system for providing efficient services to the users. In this, the routing-based request prediction system was focused to analyze and predict relevant data from network. For this reason, an enhanced clustering, distance-based similarity and retrieving mechanisms are utilized. Here, the irrelevant parameters removal and ordering are performed to preprocess the dataset. Then, DCP measure is used to select the CP by extracting the parameters from the repository of files. Also, the LS mechanism is used to form the cluster with set of nodes based on the CP. The nodes are then processed for the learning model in data learning system by using VPM prediction model.

The validation parameters compare the existing system of routing which show that DCP-VPM model improves the range of communication with the fast data transmission in CRSN than the other conventional methods. This also reduces complexity of a system represented by the reduction of number of iterations to reduce the cost value and to converge the error rate for each update of traffic level in the link path of nodes.

The CRSN nodes are clustered in such a way that the most reliable licensed channels for data transmission are chosen. To improve network performance, optimal transmission is responsible for load balancing as well as energy balancing. In addition, data packets are sent to sink nodes via a trustworthy next hop forwarder. Extensive simulations reveal that the suggested routing protocol is more reliable and energy efficient.

In the future, this type of fast optimal routing system and efficient routing system using the channel selection process can be implemented with a reduced bit size of the parameters generation system, which can improve the memory management in the parameters exchange process. Thus, an investigation on how to integrate this mechanism while minimizing end-to-end delay, interference and improving the reliability, should be part of future studies.

References

1) Pasichnyk V, Kunanets N, Veretennikova N, Rzheuskyi A, Nazaruk M. Simulation of the Social Communication System in Projects of Smart Cities. *2019 IEEE 14th International Conference on Computer Sciences and Information Technologies (CSIT)*. 2019;3. Available from: http://eiburs-ascimer.transyt-projects.com/files/05_TorregrosaMartin_AndreayMario_ConceptChallenges&Projects.pdf.
2) Li X, Xiao H, Tian J. Energy-Efficiency Maximization Based Resource Allocation for RF Energy Harvesting Underlay CRN With QoS Guarantee. *2019 IEEE 19th International Conference on Communication Technology (ICCT)*. 2019;p. 892–896. doi:10.1109/ICCT46805.2019.8947013.
3) Vakili A, Navimipour NJ. Comprehensive and systematic review of the service composition mechanisms in the cloud environments. *Journal of Network and Computer Applications*. 2017;81:24–36. Available from: https://dx.doi.org/10.1016/j.jnca.2017.01.005.
4) Du Y, Xue L, Xu Y, Liu Z. An apprenticeship learning scheme based on expert demonstrations for cross-layer routing design in cognitive radio networks. *AEU - International Journal of Electronics and Communications*. 2019;107:221–230. Available from: https://dx.doi.org/10.1016/j.aeue.2019.05.041.
5) jun Lei K, hong Tan Y, Yang X, rui Wang H. A K-means clustering based blind multiband spectrum sensing algorithm for cognitive radio. *Journal of Central South University*. 2018;25(10):2451–2461. Available from: https://dx.doi.org/10.1007/s11771-018-3928-z.
6) Sun C, Wang Y, Wan P, Du Y. A cooperative spectrum sensing algorithm based on principal component analysis and K-medoids clustering. *2018 33rd Youth Academic Annual Conference of Chinese Association of Automation (YAC)*. 2018;17. Available from: https://doi.org/10.1186/s13638-019-1338-z.
7) Magdalene AHS, Thulasimani L. Fuzzy Clustering Means (FCM) for Mitigating Spectrum Sensing Data Falsification (SSDF) Attack in Cognitive Radio Networks. *2017 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*. 2017.
8) Hossain MA, Schukat M, Barrett E. Enhancing the Spectrum Sensing Performance of Cluster-Based Cooperative Cognitive Radio Networks via Sequential Multiple Reporting Channels. *Wireless Personal Communications*. 2021;116(3):2411–2433. Available from: https://dx.doi.org/10.1007/s11277-020-07802-4.
9) Zuo P, Wang X, Linghu W, Sun R, Peng T, Wang W. Prediction-Based Spectrum Access Optimization in Cognitive Radio Networks. *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. 2018;p. 1–7. doi:10.1109/PIMRC.2018.8580726.
10) Supraja P, Pitchai R, Raja. Spectrum Prediction in Cognitive Radio with Hybrid Optimized Neural Network. *Mobile Networks and Applications*. 2019;24(2):357–364. Available from: https://doi.org/10.1007/s11036-017-0909-7.
11) Liu Z, Li C. On spectrum allocation in cognitive radio networks: a double auction-based methodology. *Wireless Networks*. 2017;23(2):453–466. Available from: https://doi.org/10.1049/iet-net.2017.0264.
12) Harris A, Karachewski M, Schnabel N. Vehicle to Everything Communication using VLC. *Electrical Engineering Senior Theses*. 2018;44. Available from: https://scholarcommons.scu.edu/elec_senior/44.
13) Xu C, Song C, Zeng P, Yu H. Secure resource allocation for energy harvesting cognitive radio sensor networks without and with cooperative jamming. *Computer Networks*. 2018;141:189–198. Available from: https://dx.doi.org/10.1016/j.comnet.2018.05.026. doi:10.1016/j.comnet.2018.05.026.
14) Yao W, Yahya A, Khan F, Tan Z, Rehman AU, Chuma JM, et al. A Secured and Efficient Communication Scheme for Decentralized Cognitive Radio-Based Internet of Vehicles. *IEEE Access*. 2019;7:160889–160900. Available from: https://dx.doi.org/10.1109/access.2019.2945610.
15) Kang S, Joo C, Lee J, Shroff NB. Pricing for Past Channel State Information in Multi-Channel Cognitive Radio Networks. *IEEE Transactions on Mobile Computing*. 2018;17(4):859–870. Available from: https://dx.doi.org/10.1109/tmc.2017.2740931.
16) Bouihi B, Bahaj M. Ontology and Rule-Based Recommender System for E-learning Applications. *International Journal of Emerging Technologies in Learning (iJET)*. 2019;14(15):4–4. Available from: https://dx.doi.org/10.3991/ijet.v14i15.10566.
17) Liu Z, Zhao M, Yuan Y, Guan X. Subchannel and resource allocation in cognitive radio sensor network with wireless energy harvesting. *Computer Networks*. 2020;167:107028–107028. Available from: https://dx.doi.org/10.1016/j.comnet.2019.107028.
18) Zareei M, Vargas-Rosales C, Hernndez RV, Azpilicueta E. Efficient Transmission Power Control for Energy-harvesting Cognitive Radio Sensor Network. *2019 IEEE 30th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC Workshops)*. 2019. doi:10.1109/PIMRCW.2019.8880825.
19) George R, Mary TAJ. Review on directional antenna for wireless sensor network applications. *IET Communications*. 2020;14(5):715–722. Available from: https://dx.doi.org/10.1049/iet-com.2019.0859.
20) Prajapat R, Yadav RN, Misra R. Energy-Efficient k-Hop Clustering in Cognitive Radio Sensor Network for Internet of Things. *IEEE Internet of Things Journal*. 2021;8(17):13593–13607. Available from: https://dx.doi.org/10.1109/jiot.2021.3065691.
21) Vishnu JB, Bhagyaveni MA. Opportunistic transmission using hybrid sensing for Cognitive Radio Sensor Network in the presence of smart Primary User Emulation Attack. *International Journal of Electronics*. 2020;108:1–15. Available from: https://dx.doi.org/10.1080/00207217.2020.1837254.
22) Tripathi Y, Prakash A, Tripathi R. An Optimum Transmission Distance and Adaptive Clustering Based Routing Protocol for Cognitive Radio Sensor Network. *Wireless Personal Communications*. 2021;116(1):907–926. Available from: https://dx.doi.org/10.1007/s11277-020-07745-w.
23) Stephan T, Al-Turjman F, Joseph KS, Balusamy B, Srivastava S. Artificial intelligence inspired energy and spectrum aware cluster based routing protocol for cognitive radio sensor networks. *Journal of Parallel and Distributed Computing*. 2020;142:90–105. Available from: https://dx.doi.org/10.1016/j.jpdc.2020.04.007. doi:10.1016/j.jpdc.2020.04.007.
24) Gatate V, Agarkhed J. Energy preservation and network critic based channel scheduling (EPNCS) in cognitive radio sensor networks. *International Journal of Information Technology*. 2021;13(1):69–81. Available from: https://dx.doi.org/10.1007/s41870-020-00523-8.
25) Sheelavant K, R S, V CK. Optimal Routing and Scheduling for Cognitive Radio Sensor Networks using Ensemble Multi Probabilistic Optimization and Truncated Energy Flow Classification Model. *International Journal of Engineering Trends and Technology*. 2021;69(9):168–178. Available from: https://dx.doi.org/10.14445/22315381/ijett-v69i9p221.