Journal of Theoretical and Applied Information Technology
15th January 2021. Vol.99. No 1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A NOVEL PENDING INTEREST TABLE SHARING SCHEME USING NEURO FUZZY LOGIC FOR NAMED DATA NETWORKING COMMUNICATION

BASSAM A. Y. ALQARALLEH [1], MALEK Z. ALKSASBEH [2], AHMAD H. AL-OMARI [3]

[1]Computer Science Department, Faculty of IT, Al-Hussein Bin Talal University (AHU), Jordan

[2]Computer Information System Department, Faculty of IT, Al-Hussein Bin Talal University (AHU), Jordan

[3]Computer Science Division, Faculty of Science, Northern Border University, Saudi Arabia

E-mail: [1]alqaralleh@ahu.edu.jo, [2]malksasbeh@ahu.edu.jo, [3]kefia@yahoo.com

ABSTRACT

At present times, Information-Centric Networking (ICN) becomes familiar as a distinct standard for next-generation Internet and exhibits the significance of restoring the present host-centric model. The data transmission in ICN is based on the name of the contents instead of the addresses of the host. Besides, named data networking (NDN) is another hot research area, which permits the user to request data with no earlier details relevant to the hosting entity. Though earlier studies on NDN offers mobility and security over the classical Internet, it suffers from the problem of pending interest table (PIT) management. Therefore, a proficient PIT management strategy is essential for the utilization of PIT memory space. For effective management of the existing PIT memory space and improvise the cache usage, a novel PIT sharing strategy using neuro fuzzy logic called NFPIT is presented. In order to select an optimal friendly node (FN) for the requestor node (RN) which has less amount of PIT space, neuro fuzzy method will be executed. In addition, we employ a deep learning model by employing the Convolution Neural Network (CNN) for rule generation. The proposed method is simulated using NS3 simulator and the simulation outcome has been experimented under different aspects. The experimental values ensure the superiority of the NFPIT method by achieving a maximum cache hit ratio of 56.4%, content diversity of 67% with less content delivery time of 7.26ms.

Keywords: *Information-Centric Networking, Named Data Networking, Pending Interest Table Sharing, Neuro Fuzzy, Cache Hit Ratio.*

1. INTRODUCTION

In the earlier days, the Internet has offered the communication media between two host systems that allowed the user to receive data from dedicated servers. Next to the development of TCP/IP protocol stacks [1], packet switching method is applied to enable the end-user to transmit various kinds of data such as document, audios, as well as video packets through the web. In spite of the reality that the Internet supported flexible nature, the present changes in the environmental application, user requirement, along with the usage pattern have drastically forced it. Presently, the uprising content-centric application areas such as social media, online shopping, YouTube [2], Netflix [3], and so on allow the user to communicate text, images, audios, and videos. These applications occupy half of the global data traffic. The current need for the creation and usage of user-generated content (UGC) is weakening the Internet because it does not contain the recently developed content dispersion approaches [4]. Most of the application data delivery scheme is related to the idea of what data is desired regardless of its location. In addition, offering mobility and security is not an inherent characteristic on the Internet the same as it gives many patches or add-on options which could be ineffective in different scenarios [5]. These difficulties have led to the development of an efficient model to the Internet that could intrinsically follow the content-centric data transmission. Amongst the various funded works of developing content-based future Internet model, Named Data Networking (NDN) is evolved as a fascinating solution [3] that dealt with application provoked variable length, location-independent names to search and retrieve contents for an intended user autonomous of the hosting entities.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

NDN is an Internet framework that is the probable adversary towards the architecture of TCP/IP. It becomes the trending research topic at present times and large number of studies have been done in this domain. Despite source and destination addresses, an NDN packet holds the data packet. With no information about the data location, the consumer of the data shows interest in the ideal data names outline. The interest has been satisfied by the router, which is delimited to names by in-between data repositories, cryptographic signatures, original data producers, or out of router caches. For communication, the implications of exclusive content name allow the router to pursue the states of the packet that comprise numerous functions unlike the IP routers. While developing the application, the fundamental modifications produce many novel chances and various logical problems, privacy and security, forwarding, and network routing. Through exchanging two packet types named Data packet (D_pkt) and Interest packet (I_pkt), the transmission of data takes place in NDN. These two packets have a name that searches for a data element that might be transmitted in one D_pkt. Each NDN router has a Pending Interest Table (PIT), Content Store (CS), and Forwarding Information Base (FIB) [4]. Through exchanging D_pkt and I_pkt, NDN communication is carried out.

A user transmits I_pkt to the requisition of information. Then, it is forwarded by the router by employing data names, and the creation or response to the D_pkt, that pursues the same I_pkt pathway, however, in the opposite manner. Every router manages the state data of unfulfilled interests in this process. This state data merged with the conversation of D_pkt as well as I_pkt and allows the NDN routers for exploring different forwarding paths, loop detection, all forwarding plane and examines the data retrieval performance. To satisfy future requests, CS is a temporary cache of content storage. If the content exists, it finds the CS and transmits the D_pkt while an I_pkt enters the node. Caching operation is performed through the NDN router, to make a duplicate copy the D_pkt that passes by CS until it is replaced by the novel content to conserve bandwidth and also to enhance sharing possibility and to minimize retrieval time. Through precise name matching procedure, the search in CS can be carried out. Subsequently, for each incoming I_pkt, PIT holds the entry until its life span gets expired or its corresponding D_pkt arrived. To forward D_pkt towards the requested user, they are used. Through appropriate name matching, the search procedure of PIT entries can be carried out. The FIB contains relative data and the next hop of each reachable destination name prefixes. Using the routing protocols, it is populated and used to forward I_pkt upstream.

To recognize the correct matches, a searching task of the respective content in CS occurs while an I_pkt from user U attains at NDN router. With the producer's key signature, the NDN router transmits the user contents through D_pkt while similar content is available. The received interface I_pkt would append interface(s) inventory called as interest aggregation while a match is established within PIT entry. Each proposed user receives a D_pkt copy, upon D_pkt receiving as a result. For an arriving I_pkt, it will be forwarded to perform the longest prefix match (LPM) and router's FIB, while no PIT entry is accessible. The entire PIT entries would be found for searching a CN match, upon the reception of D_pkt at the NDN router. The D_pkt is passed to each interface in the incoming interface(s) list while corresponding PIT entry is proposed. Therefore, within CS, the PIT entry is eliminated, and content is saved depending on local caching policies fundamental for serving future requirements. The D_pkt will be dropped while zero equal entry is presented.

NDN nodes might comprise the sum of the rest of the PIT memory in a few circumstances that limit the nodes to store high interests. This paper presents a novel PIT sharing strategy based on neuro-fuzzy called as NFPIT technique. The proposed method shares the PIT space of the adjacent NDN node which has higher remaining PIT space with its adjacent NDN node which has lower remaining PIT space. To choose the optimal friendly node (FN) for the requestor node (RN) which has less amount of PIT space, the neuro fuzzy method will be executed. The NFPIT will offer a set of optimal FN node for the RN. The proposed NFPIT method will balance the amount of PIT memory among the nodes that exist in the network.

The succeeding portion of the study is planned as follows: Section 2 summarizes the work relevant to the proposed NFPIT method. Section 3 introduces the NFPIT method and Section 4 validates the NFPIT method. Section 5 draws a conclusion.

2. BACKGROUND INFORMATION AND RELATED WORK

For every I_pkt, an NDN router comprises of a PIT entry as the NDN forwarding is considered to be stateful. As shown in Figure 1, five fields such as nonce, outgoing interface(s), CN, timer, and incoming interface(s) are contained in each PIT entry. In terms of CN, Nonce is in merger and 4-octet long byte string, as it searches I_pkt separately and

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

it prevents forwarding of replica packet. A timer is linked over an equivalent PIT entry when I_pkt is passed by the NDN router. In case, when no D_pkt is established till the timer expires, the respective PIT entry is removed. Deletion of a serviced entry, insertion of new entry, and updates existing entry are the three main PIT operations. Each operation needs to find the PIT for the subsequent entry availabilities. Extensively, PIT design is segmented into two phases: PIT placement and PIT data structure as shown in Figure 1. Bloom filter [6], trie [7] and hash table [8, 9] are data structures used by a few studies to use PIT over resource-limited routers to decrease access time and memory consumption.

2.1 PIT data Structure

Depending on Encoded Name Prefix Trie (ENPT) performs encoding of CN elements in terms of 32-bit integers for diminishing PIT access time and PIT size, Name Component Encoding (NCE) is projected. For component encoding, NCE uses a hash function. The encoded name will be appended with ENPT, next to component encoding. The important disadvantage of ENPT is to ease n^{th} element code generation depending on the $(n-1)^{th}$ element code which might diminish the results of the model. Using d-left open-addressed hash table (DHT), the PIT data structure is proposed [10]. DHT performs well over NCE, hash table, and Counting Bloom Filter (CBF) [11-14]. Li et al. [15] make use of mapping bloom filter (MBF) to highly decrease PIT time, NCE technique, and on-chip memory. Mapping and querying of set elements are supported by MBF in memory additionally diminishes the on-chip memory cost. For core routers, Yuan et al. [14] aimed at the compacted storage of PIT design as it highly depends on the performance of the network. For using fixed-length fingerprints, d-left hash table is projected by authors. To examine the maximum and average PIT size, Carofiglio et al. [16] projected a PIT dynamics analytical model.

2.2 PIT Position

Even though PIT data structure diminishes the consumption of storage space at a certain point in high associating rates, storage requirements might exceed the certain memory chip dimension. For each outward/inward router interface to manage the stakeholder need, it is required to partition PIT. Similar to input–output line-cards and output line-card, 2 PIT placements have been proposed [17]. At each outgoing interface, distinguished PIT is located in the output line-card. I_pkt creates its entry on the output line-card and it is passed. Authors in [10] projected an input line-card PIT placement where separated PIT is positioned. This technique fails to offer loop detection and Interest aggregation, it requires a finding of the whole PIT accessible on diverse inward interfaces while D_pkt enters. A third-party PIT placement has been introduced when PIT is placed in each input-line-card. For certain content requests received, a third party selects the PIT using hashing [18]. There is no need for line-card because the same process is returned for D_pkt CN. This technique provides Interest aggregation, multi-path forwarding, and loops identification. For controlling, it takes switches additionally mutually D_pkt and I_pkt.

Name	List of Nonce	List of Incoming Interfaces		List of Outgoing Interfaces	
CN	Nonce	Interface ID	Timer	Interface ID	Send Time

Figure 1: Structure Of PIT Entry

3. PROPOSED ALGORITHM

In the proposed work, the FN is chosen based on the characteristics of the NDN node present around the RN. In addition, we employ a deep learning model by employing the Convolution Neural Network (CNN) for rule generation. CNN in this work employs two hidden layers subjected for processing, one output layer for the result, and one input layer for inputs. The convolution layer is one of the two hidden layers. From the previous and current network trace data, a neural network (NN) is used. The historical data are employed for primary training. Through the implication of present data, the weights are tuned and using the fuzzy rules, they are executed through a fuzzy inference system. By the CNN training, rules are formed which is used to search a highly optimal route that needs effective PIT memory utilization.

Through communicating the data collected from the NDN nodes, testing is performed. The four parameters used to select the FN node are distance between the mobile NDN nodes, PIT remaining memory, node mobility, and node degree. Figure 2 demonstrates an overview of the presented NFPIT method. It consists of a set of 256 rules from hidden layers, 4 inputs and 1 output from the input and output layers respectively.

The projected framework comprises of main modules like Neural Network module for training and testing, Neuro Fuzzy rule manager, Base station, Decision Manager, IoT based mobile NDN nodes, Neuro-Fuzzy inference system and Rule base. To obtain effective PIT memory utilization, the entire modules are combined to perform decision making process. For sharing PIT memory, FN is chosen based on some criteria. In prior to sharing the data to FN, it should be ensured that the FN can satisfy the requirements of the RN. By employing Equation (1), this task estimates the node degree. There exists a low probability for an RN to merge into FN when the node degree is high.

$$NDN_{degree} = \frac{Numberof\ adjacent\ NDN\ nodes}{Totalnumberof\ NDN\ nodes} \quad (1)$$

For instance, if the node count is 100 within the network and the number of nearby nodes is 10, then NDN_{degree} is considered as 0.1.

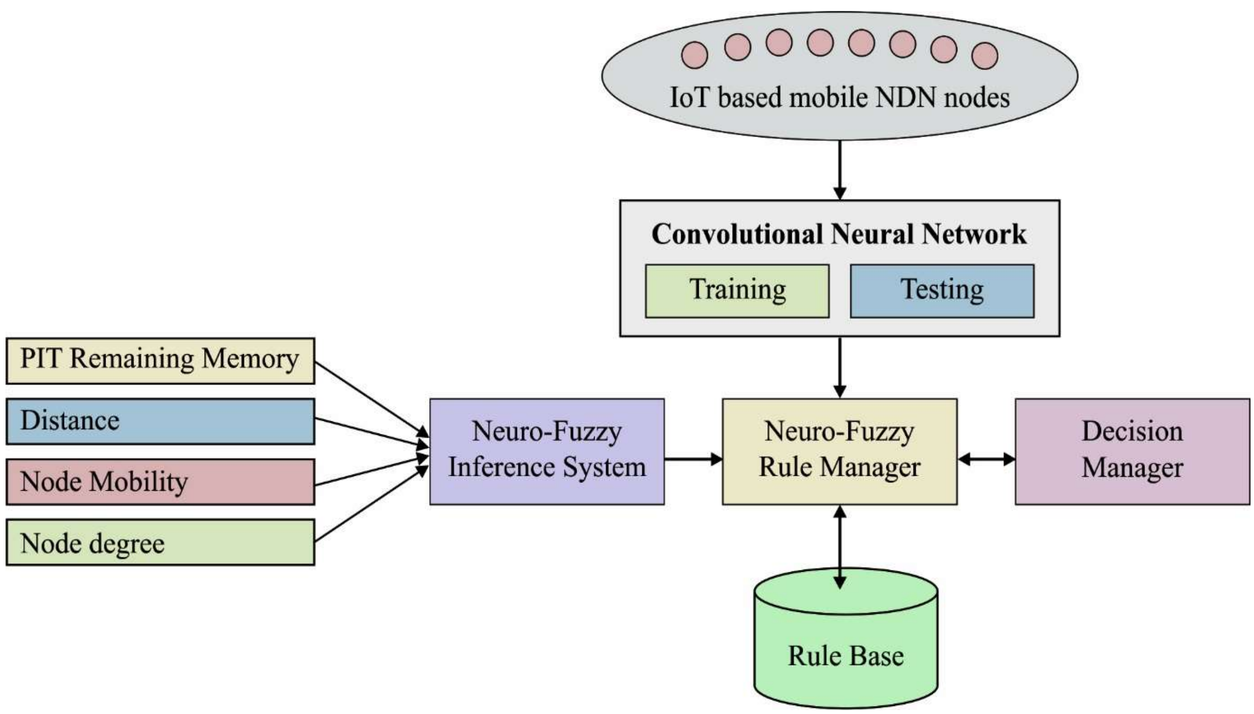

Figure 2: Architecture Of The NFPIT

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 1: Fuzzy Rule Base Table

Rule	Input variable				Output variable
	Remaining PIT memory	Node Degree	Distance	Node mobility	Prob. of FN
1	L	L	F	L	L
2	L	L	F	L	VL
3	L	L	F	L	VL
.					
20	L	L	C	L	L
-					
28	L	M	F	L	RL
57	L	H	F	L	L
61	L	H	F	L	RL
-					
63	L	H	F	L	VL
-					
82	M	L	F	M	M
-					
109	M	M	F	M	RH
-					
127	M	M	C	M	RL
-					
136	M	H	F	M	RH
-					
190	H	M	F	H	H
-					
217	H	M	F	H	VH
218	H	H	F	H	RH
219	H	H	F	H	M
223	H	M	F	H	RH
224	H	H	F	H	M
225	H	H	F	H	RL
226	H	M	M	H	VH
227	H	H	M	H	H
228	H	H	M	H	RH
232	H	M	M	H	M
233	H	H	M	H	M
234	H	H	M	H	M
-					
238	H	M	C	H	H
239	H	H	C	H	RH
240	H	H	C	H	M
241	H	M	C	H	RH
-					
256	H	H	C	H	M

3.1 Neuro-Fuzzy Membership Functions

For performing training and communication links structure in IoT based mobile NDN nodes, CNN is used in this work. For searching the FN with nearby distance to RN, this method allows the network to use neuro-fuzzy rules. With a view to producing decision rules, NFIS employs trapezoidal and triangular membership functions with CNN. The neuro-fuzzy rule system employs fuzzy membership functions that are demonstrated in Equations (2) and (3) respectively.

$$\mu_{A1}(z) = \begin{cases} 0 & z \le a1 \\ \dfrac{z-p1}{q1-p1} & p1 \le z \le q1 \\ \dfrac{r1-z}{r1-q1} & q1 \le z \le r1 \\ 0 & r1 \le z \end{cases} \tag{2}$$

$$\mu_{A1}(z) = \begin{cases} 0, & z \le a1 \\ \dfrac{d2-x}{d2-r2}, & r2 \le z \le d2 \\ 1, & q2 \le z \le r2 \\ \dfrac{d2-x}{d2-r2}, & r2 \le z \le d2 \\ 0, & r1 \le z \end{cases} \tag{3}$$

The attributes x and z demonstrate the actual and fuzzy distances respectively. By using the membership functions, the fuzzy distance value is measured. Node mobility refers to the average speed of the NDN node moves in the network.

3.2 Neuro Fuzzy Rules

Primarily, for weight adjustment, we have done CNN training with the fuzzy method in this work. In addition, because of the CNN [19] nature, the Mamdani Fuzzy inference model has been employed [20]. A set of 4 input variables are comprised of the projected NFPIT method and every input is comprised of 3 levels. The levels employed for every fuzzy variable is expressed as:

- PIT remaining memory – Low (L), Medium (M), High (H)
- Distance – Close (C), Medium (M), Far (F)
- Node degree –L, M, H
- Node mobility - L, M, H
- FN probability – Very low (VL), Rather low (RL), Low L, M, H, Rather high (RH), very high (VH)

By employing a set of IF-THEN rules$4^4 = 256$, the probability of FN is determined. To demonstrate the member choice levels, the trapezoidal and triangular functions are employed. Table 1 provides the set of IF-THEN rules employed in fuzzy logic in the projected approach.

De-fuzzification step is the final step in the procedure of fuzzy rule-based inference. With respect to fuzzy values, the task of de-fuzzification is implied to gain a crisp output rate. The Center of Area method is extensively employed amongst the available de-fuzzification methods; so, this work uses this method and is expressed in Equation (4):

$$COA = \frac{\int \mu_A(z).zdz}{\int \mu_A(z)dz} \tag{4}$$

The FN selection process for PIT sharing mechanism has been projected in this work that performs an optimal selection of FN for the particular RN. Using the proposed method, the RN searches for the nearest FN and manages to find the optimal FN which satisfies the needed parameters. In circumstances like bandwidth availability, congestion status, and traffic level, the network is trained by CNN. For effective communication between FN and RN for PIT memory sharing, the presented method employs CNN and fuzzy rules have been projected in this work.

The steps of the projected method are given as follows.

- **Step 1**: Interpret the location (x_i, y_i) and PIT memory space of the NDN nodes, NDN $_i$, $i = \{1, 2\ldots, n\}$.
- **Step 2**: Towards the entire neighboring nodes, transmit "HELLO" packets from the RN, and find the node distances amongst the adjacent nodes.
- **Step 3**: Compute the node degree and node mobility of the nearby nodes of the RN.
- **Step 4**: Employing RN as the coordinator, the selection of FN is performed for every NDN node by assuming the node's PIT memory, distance, node degree, and node mobility.
- **Step 5**: Perform discovery of route through searching the shortest path from every node towards the FN by the RN.
- **Step 6**: Once the FN is determined, the overflowing I-pkts at the RN will be transmitted to the FN.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

- **Step 7**: FN receives the I-pkts sent by RN and stores it until the corresponding D-pkt is received.
- **Step 8**: upon the requirement of the I-pkts stored in FN, the RN made a request to FN and accesses the I-pkts successfully.

By doing this, the presented NFPIT method successfully achieves the PIT sharing mechanism by selecting the optimal FN for the RN.

4. PERFORMANCE VALIDATION

4.1 Implementation Setup

The presented PIT sharing mechanism is validated by the use of ndnSIM, which is an NS-3 based NDN simulation tool [21]. For validation, a 6-node Sprint PoP topology is given in Figure 3. Every gateway router is linked to a creator, and every access router is linked to a set of 3 users. It is assumed that every router holds a CS that can save 100 diverse contents with the content size of 1024 bytes. Every node has a PIT to store the interest packets per second for every user based on the uniform or Poisson distribution, and the reputation of I_pkts follow Zipf distribution [22] with the object popularity of $\alpha = 0.7$. The Zipf distribution is commonly employed for modeling the Internet traffic pattern [22]. The aim of the proposed method is limited to sharing the PIT space and is limited to its dedicated one-hop neighbors. The parameter setting is given in Table 2. And, Least Recently Used (LRU) and Least Frequently Used (LFU), LRU with summary sharing (LRU-SS) and LFU with summary sharing (LFU-SS) [23] are employed to compare the results of the presented method. The results are compared with 5 diverse methods as mentioned above.

Table 2: Parameter Settings

Parameter	Standard values
Request rate	100 Interest/s
Node capacity	100 contents
Content sizes	1KB
Catalog sizes	900

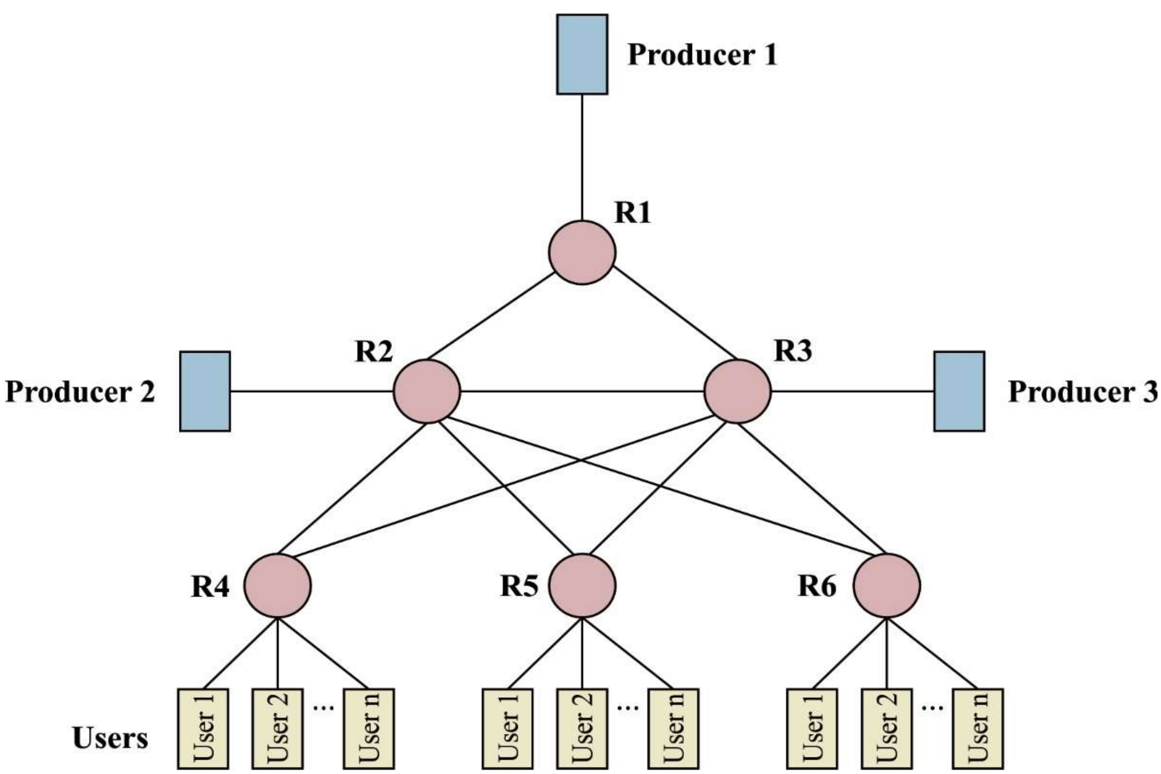

Figure 3: Sprint PoP Springfield Topology

4.2 Performance Metrics

A total of three evaluation parameters are employed to validate the results of the presented method and the parameters [18, 24] are cache hit ratio, content diversity, and content delivery time. Rather than analyzing the cache hit ratio of a separate cache, the cache hit ratio is assumed as the rate of requests made by every cache as equated in Eq. (5).

$$Cache\ hit\ ratio = \frac{\sum_{i=1}^{N} hits_i}{\sum_{i=1}^{C} R_i} \quad (5)$$

where N is the router count, C is the number of users, and R_i is the number of interest packets generated by the user. The content diversity is computed as the ratio of the summated length of diverse cached contents to the summated length of the whole cache and is shown below.

$$Content\ Diversity = \frac{len\ (\cup_{i=1}^{N} CS_i)}{\sum_{i=1}^{N} len\ (CS_i)} \quad (6)$$

Where CS_i represents the content, which is saved in CS of router i. The content diversity attains higher value in case of no replica packet present in all the caches. In case of the absence of a replica packet in the network, the high cache hit ratio is also attained and the network can manage with the drastic modifications in the request familiarity. The average content delivery time is related to the routing of I-pkt and D-pkt. The delivery time is the amount of time needed by the Data packet reaches the user from the second the interest packet has been created by the user. So, the average content delivery time can be determined as follows.

$$Avg.Delivery\ time = \frac{\sum_{i=1}^{C}\sum_{j=1}^{R_i}(t(Data_{i,j}) - t(Interest_{i,j}))}{\sum_{i=1}^{C} R_i} \quad (7)$$

The aim of this PIT sharing mechanism is to effectively utilize the PIT memory space and store the excess interest packets in the friendly node. It leads to a high cache hit ratio and low content delivery time.

4.3 Results Analysis

For validating the scalability of the NFPIT method, a series of experiments are conducted to study the effect of content catalog size[24]. Figures 4-6 and 7-9 show the performance analysis of content catalog size during the generation of I_pkts by uniform as well as Poisson distribution, correspondingly. Figures 4-6 depict the results attained by the cache hit ratio, content diversity, and average delivery time, correspondingly, under the uniform distribution. Figures 7-9 illustrate the outcome of the Poisson distribution case. The proposed NFPIT method performs well with LFU as well as LRU, with respect to the number of contents in the network.

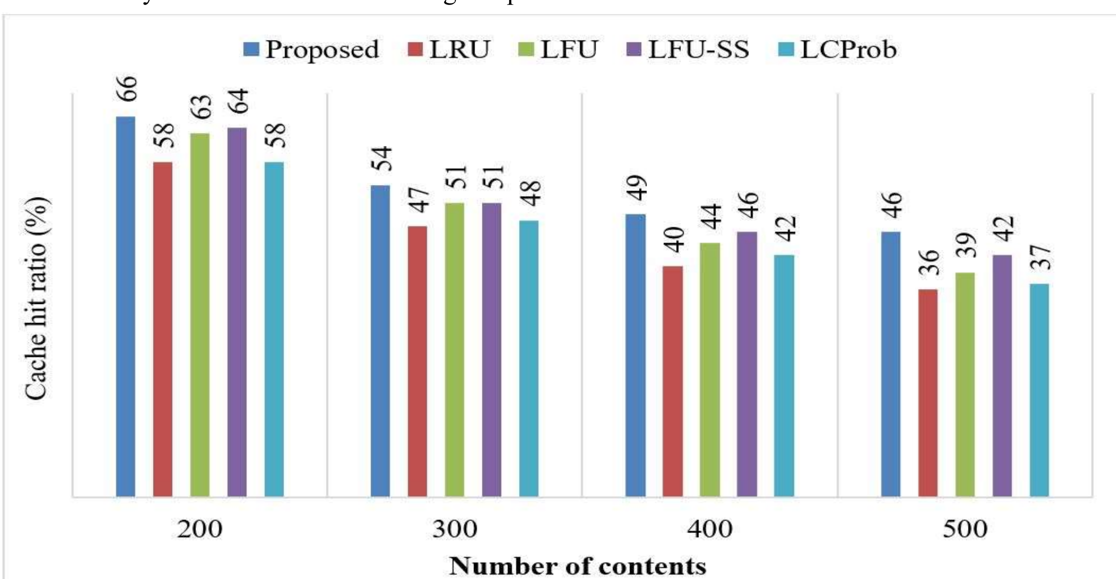

Figure 4: Comparison Of Different Methods In Terms Of Cache Hit Ratio (Uniform Distribution)

Figure 4 shows the comparison of NFPIT method with various approaches with respect to cache hit ratio. As shown above, it is apparent that the presented NFPIT approach shows effective results under a varying number of contents. For instance, under 200 contents, the NFPIT method attains a maximum cache hit ratio of 66% whereas the existing LRU and LUProb methods attain a minimum cache hit ratio of 58%. The other LFU and LFU-SS methods obtain better results compared to LRU and LUProb with the cache hit ratio of 63% and 64% respectively. But it does not outperform the NFPIT method. At the same time, under 300 contents, the methods LRU and LCProb gain more or less than the same performance and it is also showed the least performance over the compared methods. LFU and LFU-SS attain same performance of 51% of cache hit ratio whereas the proposed method shows superior performance by attaining 54% of cache hit ratio. The similar kinds of results are exhibited under 400 and 500 categories in which the proposed method shows superior performance. Therefore, the proposed method exhibits an enhanced cache hit ratio under a varying number of contents.

Figure 5 demonstrates the comparison of various techniques by means of content diversity. Under 200 contents, LRU is the worst performer as it attains a low rate of 46% of content diversity rate. LFU and LCProb attain enhanced performance than the above method by gaining 55% and 53% of content diversity rate. But it does not exceed the method LFU-SS which reaches 61% of content diversity rate. The projected NFPIT technique attains 66% which is higher than all the other methods by means of content diversity rate. Under 500 contents category, LRU shows a reduced content diversity rate of 50% and it is poorly performed over the compared methods. LFU and LCProb attained the same content diversity rate of 58% but it is better than LRU. The method LFU-SS shows a good content diversity rate of 65% which is superior among the compared methods, but it fails to outperform the projected method which attains 70% of content diversity rate.

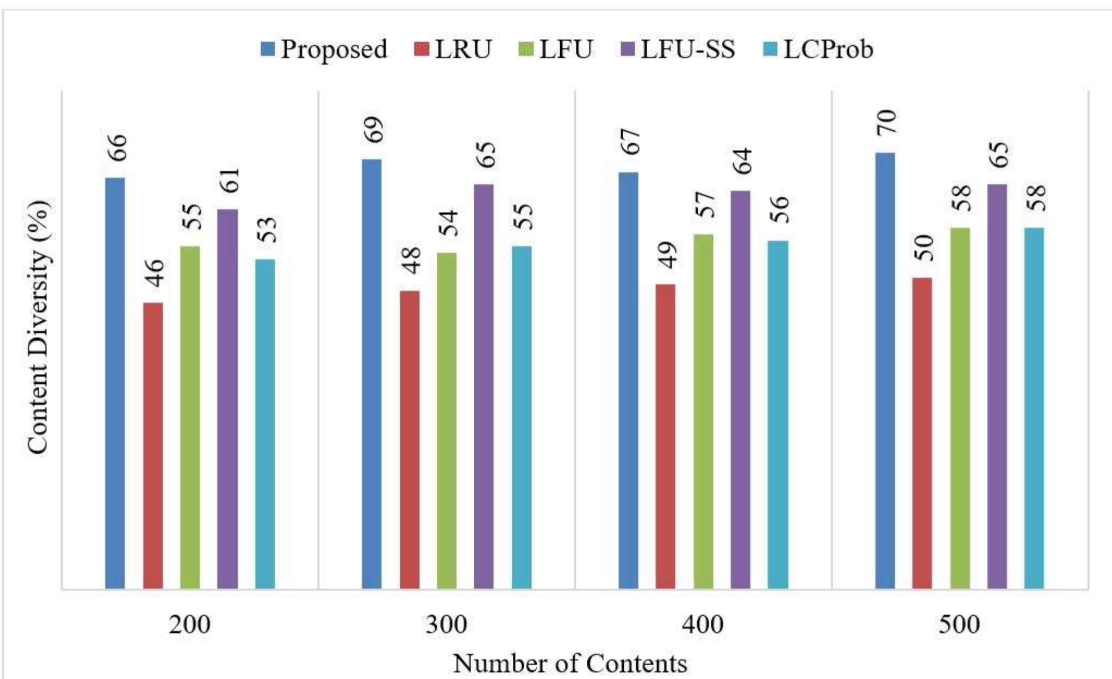

Figure 5: Comparative Analysis Of Various Approaches With Respect To Content Diversity (Uniform Distribution)

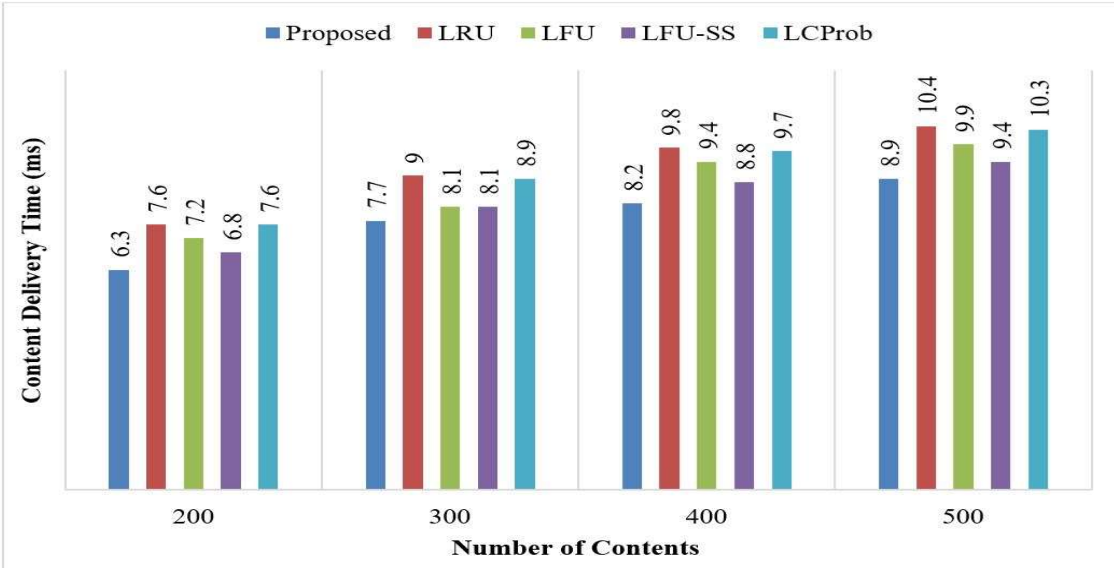

Figure 6: Comparison Of Different Models With Respect To Content Delivery Time (Uniform Distribution)

In terms of content delivery time, the techniques are compared under various categories of a varying number of contents are given Figure 6. Content delivery time should be low for better performance. Under 300 contents category, LRU shows the worst performance by attaining a high content delivery time of 9ms. The method LCProb requires 8.9ms of content delivery time. When taking into consideration of methods LFU and LFU-SS, it shows a similar performance of 8.1ms. The proposed method attains enhanced performance by requiring a minimum of 7.7ms as content delivery time. In 200 contents category, LRU and LCProb show similar and worst performance by the content delivery time of 8.1ms. LFU shows better performance than the above techniques by attaining the content delivery time rate of 7.2ms. The LFU-SS exhibit competitive results with the content delivery time of 6.8. But the proposed method shows superior performance by attaining content delivery time of 6.3ms. Similarly, for the category of 400 and 500 contents, identical performance is attained by the presented method.

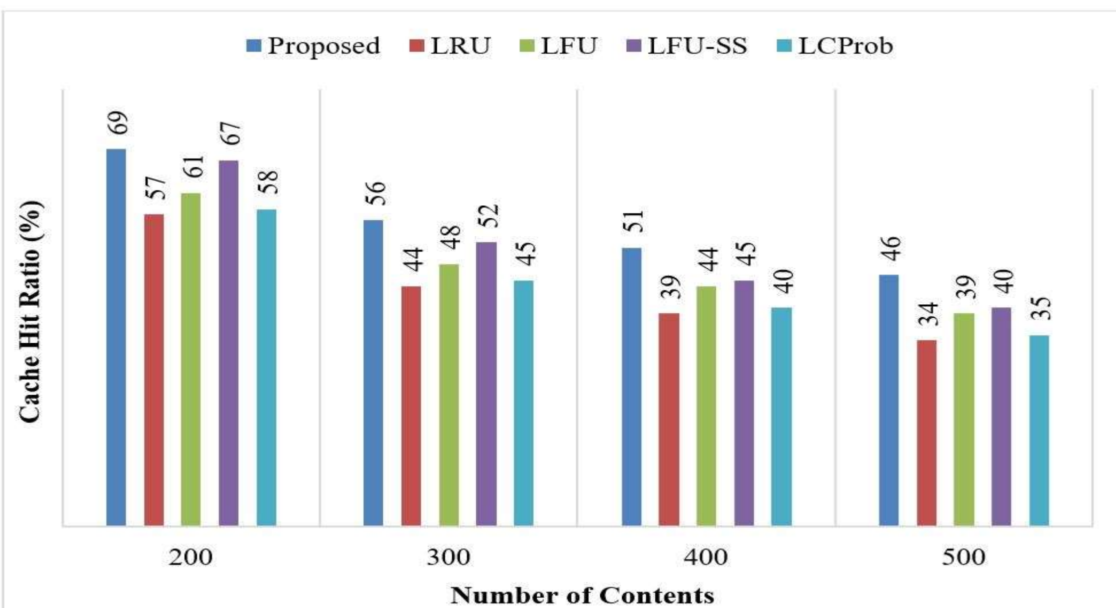

Figure 7: Comparison Of Different Methods In Terms Of Cache Hit Ratio (Poisson Distribution)

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figures 7-9 demonstrates the performance comparison of various techniques under Poisson distribution. Under the 500 contents category, cache hit ratio produced by different methods and presented methods are discussed in Figure 7. LFU and LCProb show approximately equal, as well as poor performance by gaining the cache, hit ratio of 34%, and 35% respectively. The methods LFU and LFU-SS show better performance than the above-mentioned ones by the cache hit ratio of 39% and 40% respectively. The presented method shows a superior cache hit ratio of 46% in this category.

While taking into consideration 400 contents category, LRU and LCProb methods show the worst performance with the cache hit ratio of 39% and 40% respectively. The methods LFU and LFU-SS exhibit good performance over the above methods, but the proposed method attained a superior cache hit ratio of 51% over the compared methods. At the same time for the other categories such as 200 and 300 contents, the same performance is attained through the proposed method; hence it is superior among all the other methods.

Figure 8 shows the comparison of different methods by means of content diversity. Under 200 contents, LRU is the worst performer as it attains a low rate of 46% as content diversity rate. LFU and LCProb attain enhanced performance than the above method by gaining 55% and 53% of content diversity. But it does not exceed the method LFU-SS which reaches 60% of the content diversity rate. The projected NFPIT technique attains 64% which is higher than all the other methods by means of content diversity rate. Under 500 contents category, LRU show reduced content diversity rate of 49.5%. LFU and LCProb attain content diversity rate of 55.8% and 58%, but it is better than LRU. The method LFU-SS shows a good content diversity rate of 68% which is superior among the other methods, but it fails to outperform the projected method which achieves 72% of content diversity rate.

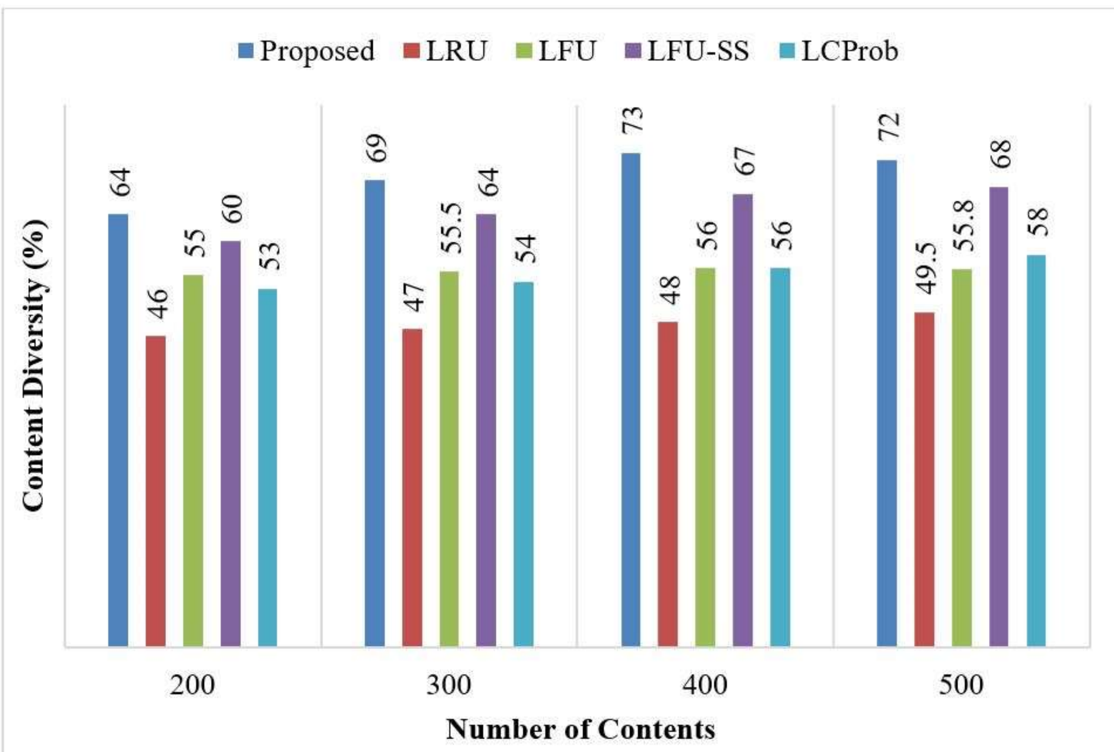

Figure 8: Comparison Of Different Methods In Terms Of Content Diversity (Poisson Distribution)

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

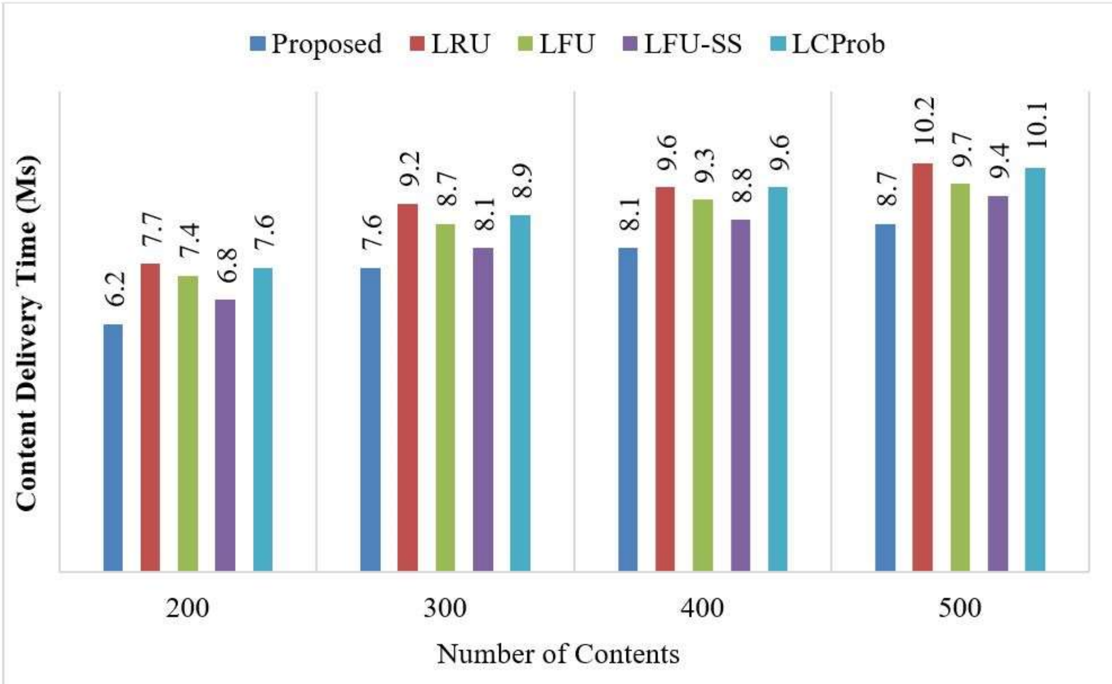

Figure. 9: Comparison Of Different Methods In Terms Of Content Delivery Time (Poisson Distribution)

By means of content delivery time, the methods are compared to one another under different categories of content count are shown in Figure 9. Under 300 contents category, LRU shows worst performance by attaining high content delivery time of 9.2ms. The method LCProb shows 8.9ms as content delivery time. The LFU-SS method requires content delivery time of 8.1ms. The proposed method attains enhanced performance by attaining 7.6ms as content delivery time. In 200 contents category, LRU and LCProb show worst performance by the content delivery time of 7.7 and 7.6ms respectively. The LFU-SS exhibits an increased content delivery time of 6.8ms. The projected method shows 6.2ms as content delivery time. Similarly, for the category of 400 and 500 contents, the same kind of performance is exhibited by the proposed method.

The NFPIT is validated on non-complete K-ary tree topology [25]. The tree depth and maximum children count of a node (K), is set as 5. The children count of every node is arbitrarily selected in the interval of [0,5]. Table 3 provides the details related to the network topology. The gateway count is identical for each topology; however, the content catalog size of the K-ary tree rises with an increase in node count in the network.

Table 3: Topology Information

Topology	Network Type	Gateway Count	Node count
1	PoP spring field	3	6
0	K-ary tree	3	26

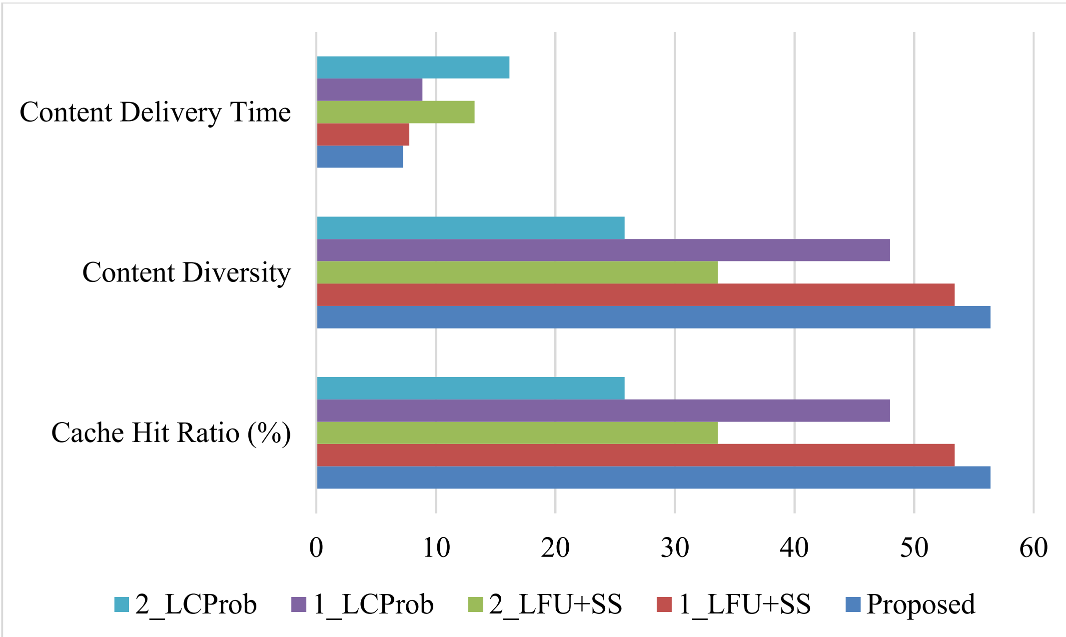

Figure 10: Comparison Of Proposed With Various Methods Under Varying Skewness Factor

Figures 10 illustrate the performance of the NFPIT algorithm with an increase in skewness factor. The efficiency of the NFPIT technique is superior to compared methods with varying skewness factor. From the figure, it is obvious that the NFPIT method attains a higher cache hit ratio, content delivery time, and content diversity. On average, the presented NFPIT method obtains a maximum cache hit ratio of 56.4%, content diversity of 67% with less content delivery time of 7.26ms. These values depicted that the presented NFPIT method effectively shared the PIT memory space of the NDN nodes in the network and thereby achieves efficient resource utilization.

5. CONCLUSION

In order to resolve the PIT storage space problem, this paper has introduced an efficient PIT sharing technique called NFPIT technique. The proposed method shares the PIT space of the adjacent NDN node which has higher remaining PIT space with its adjacent NDN node which has lower remaining PIT space. In the proposed work, the FN is chosen based on the characteristics of the NDN node present around the RN. In addition, this research employs a deep learning model by applying CNN for rule generation. The proposed method is simulated using NS3 simulator and the results are validated under several aspects. On average, the presented NFPIT method achieves a maximum cache hit ratio of 56.4%, content diversity of 67% with less content delivery time of 7.26ms. These values depicted that the presented NFPIT method effectively shared the PIT memory space of the NDN nodes in the network and thereby achieves efficient resource utilization. In the future, the performance of the NFPIT method can be improved using advanced caching mechanism and data deduplication techniques. Besides, the retrieval efficiency of the NFPIT model can be enhanced using dictionary-based coding techniques.

REFERENCES:

[1] E. G. Britton, J. Tavs, and R. Bournas, “TCP/IP: The next generation,” *IBM Systems Journal,* vol. 34, no. 3, pp. 452-471, 1995.

[2] B. Mathieu, P. Truong, W. You, and J.-F. Peltier, “Information-centric networking: A natural design for social network applications,” *IEEE Communications Magazine,* vol. 50, no. 7, pp. 44-51, 2012.

[3] L. Zhang, D. Estrin, J. Burke, V. Jacobson, J. D. Thornton, D. K. Smetters, B. Zhang, G. Tsudik, D. Massey, and C. Papadopoulos, “Named data networking (ndn) project,” *Relatório Técnico NDN-0001, Xerox Palo Alto Research Center-PARC,* vol. 157, pp. 158, 2010.

[4] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

"Networking named content," in Proceedings of the 5th international conference on Emerging networking experiments and technologies, 2009, pp. 1-12.

[5] D. Chen, and H. Zhao, "Data security and privacy protection issues in cloud computing," in 2012 International Conference on Computer Science and Electronics Engineering, 2012, pp. 647-651.

[6] H. Hu, J. Bi, T. Feng, S. Wang, P. Lin, and Y. Wang, "A survey on new architecture design of internet," in 2011 International Conference on Computational and Information Sciences, 2011, pp. 729-732.

[7] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A survey of information-centric networking," *IEEE Communications Magazine,* vol. 50, no. 7, pp. 26-36, 2012.

[8] A. Jaffri, Z. Ahmad, and M. Tahir, "Named Data Networking (NDN), new approach to future Internet architecture design: A survey," *Int. J. Informatics Commun. Technol,* vol. 2, no. 3, pp. 155-164, 2013.

[9] M. F. Bari, S. R. Chowdhury, R. Ahmed, R. Boutaba, and B. Mathieu, "A survey of naming and routing in information-centric networks," *IEEE Communications Magazine,* vol. 50, no. 12, pp. 44-53, 2012.

[10] M. Varvello, D. Perino, and L. Linguaglossa, "On the design and implementation of a wire-speed pending interest table," in 2013 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), 2013, pp. 369-374.

[11] W. You, B. Mathieu, P. Truong, J.-F. Peltier, and G. Simon, "Dipit: A distributed bloom-filter based pit table for ccn nodes," in 2012 21st International Conference on Computer Communications and Networks (ICCCN), 2012, pp. 1-7.

[12] Z. Li, J. Bi, S. Wang, and X. Jiang, "Compression of pending interest table with bloom filter in content centric network," in Proceedings of the 7th International Conference on Future Internet Technologies, 2012, pp. 46-46.

[13] D. Perino, and M. Varvello, "A reality check for content centric networking," in Proceedings of the ACM SIGCOMM workshop on Information-centric networking, 2011, pp. 44-49.

[14] H. Yuan, T. Song, and P. Crowley, "Scalable NDN forwarding: Concepts, issues and principles," in 2012 21st International Conference on computer communications and networks (ICCCN), 2012, pp. 1-9.

[15] Z. Li, K. Liu, Y. Zhao, and Y. Ma, "MaPIT: an enhanced pending interest table for NDN with mapping bloom filter," *IEEE Communications Letters,* vol. 18, no. 11, pp. 1915-1918, 2014.

[16] G. Carofiglio, M. Gallo, L. Muscariello, and D. Perino, "Pending interest table sizing in named data networking," in Proceedings of the 2Nd ACM Conference on Information-Centric Networking, 2015, pp. 49-58.

[17] H. Dai, B. Liu, Y. Chen, and Y. Wang, "On pending interest table in named data networking," in 2012 ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS), 2012, pp. 211-222.

[18] P. Uma Maheswari, P. Manickam, K. Sathesh Kumar, A. Maseleno, and K. Shankar, "Bat optimization algorithm with fuzzy based PIT sharing (BF-PIT) algorithm for Named Data Networking (NDN)," *Journal of Intelligent & Fuzzy Systems,* vol. 37, no. 1, pp. 293-300, 2019.

[19] A. Hamzah, M. Shurman, O. Al-Jarrah, and E. Taqieddin, "Energy-efficient fuzzy-logic-based clustering technique for hierarchical routing protocols in wireless sensor networks," *Sensors,* vol. 19, no. 3, pp. 561, 2019.

[20] M. H. E. Ahmadi, S. J. Royaee, S. Tayyebi, and R. B. Boozarjomehry, "A new insight into implementing Mamdani fuzzy inference system for dynamic process modeling: Application on flash separator fuzzy dynamic modeling," *Engineering Applications of Artificial Intelligence,* vol. 90, pp. 103485, 2020.

[21] C. Yi, A. Afanasyev, I. Moiseenko, L. Wang, B. Zhang, and L. Zhang, "A case for stateful forwarding plane," *Computer Communications,* vol. 36, no. 7, pp. 779-791, 2013.

[22] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and Zipf-like distributions: Evidence and implications," in IEEE INFOCOM'99. Conference on Computer Communications. Proceedings. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. The Future is Now (Cat. No. 99CH36320), 1999, pp. 126-134.

[23] S. Podlipnig, and L. Böszörmenyi, "A survey of web cache replacement strategies," *ACM Computing Surveys (CSUR),* vol. 35, no. 4, pp. 374-398, 2003.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[24] C. Bernardini, T. Silverston, and O. Festor, "A comparison of caching strategies for content centric networking," in 2015 IEEE Global Communications Conference (GLOBECOM), 2015, pp. 1-6.

[25] W. K. Chai, D. He, I. Psaras, and G. Pavlou, "Cache "less for more" in information-centric networks (extended version)," *Computer Communications,* vol. 36, no. 7, pp. 758-770, 2013.