Robust Federated Learning-based Content Caching over Uncertain Wireless Transmission Channels in FRANs

Sanaullah Manzoor, *Student Member, IEEE,*, and Adnan Noor Mian, *Member, IEEE,*

***Abstract*—Content caching has been considered as an effective way to offload contents at network edge in order to alleviate back-haul load. Recently, federated learning (FL) based edge caching has gained a lot of popularity due to its prominent features of data privacy, distributed mode of operation, and scalability. However, these FL-based schemes ignore the behavior of the communication channels during the federated weight averaging procedure. In this paper, we introduce a novel robust federated learning-based content caching approach for fog radio access networks (F-RANs) that mitigates the effect of communication channel noise. In our proposed robust FL approach, each cell employs a deep neural network (DNN)-based model to predict users' future files rating score based on user and file contextual information and shares its learned weights to the fog server. The fog server is responsible for global weight averaging. Prior to FL weight averaging fog sever feed incoming local model weights to a generative adversarial neural network (GANs) model which differentiates between noisy and actual federated weights, and passes only actual weights based on the distribution of the weight matrices. Extensive simulations have been carried out to validate the performance of our proposed approach. Results show that the GAN-aided federated model yields 23.1% more prediction accuracy as compared to the federated noisy model without GANs based noise mitigation.**

Keywords:
Proactive caching, federated machine learning, fog radio access networks, quality of service

I. Introduction

Nowadays, a large number of wireless devices are being connected to the Internet ranging from smartphones, pods, medical devices, smartwatches, and laptops. With the evolution of wireless networks ($6G$) with small and macro cell based heterogeneity, and Internet of Things (IoTs) support, the number of device connections are rising at an exponential pace, and as a result, we witness a massive surge in Internet traffic growth. According to Cisco report, in near future, 50 billion devices are going to be connected to the Internet, and on-average each user will be generating nearly 60 GB data traffic per month [1], [2]. Furthermore, it is anticipated that almost 82% of Internet traffic will be originated from the data-hungry streaming applications such as *NetFlix*, *YouTube*, and *Spotify* [3]. This huge amount of traffic flow is posing great challenges to existing cellular and IoT infrastructure and may lead to core network overloading. On the bigger picture, on one side users' demand for the premium quality of service (QoS) increasing, while on the other side mobile network operators may not able to incur large investments to buy more bandwidths for coverage and capacity. Thus, there is a need of an urgent and cost-effective solutions to alleviate the core traffic burden by making full use of edge caching, information processing, and communication capabilities of fog radio access networks (FRANs).

Content caching at network edge has been considered as a promising solution to alleviate core network burden, and it also improves users' QoS by full filling requests with lower service delays [4]. When a user requests some content such as a video or text file, the service provider firstly checks content locally if it is available at edge cache the content is delivered otherwise, a cache miss occurs and the content has to be fetched from the remote data center which causes higher delay cost. In order to establish proactive content caching users' future content demands are learned and the future contents are then fetched before being asked. Traditional content caching approaches such as least recently used (LRU), and first-in-first-out (FIFO) based schemes are unable to incorporate the content popularity information and result in lower cache hits [5]. Some recent edge caching schemes [6]–[9] operate in a fully centralized manner where the central server takes caching decision and broadcasts to associated nodes. These centralized approaches require the uploading of large volumes of users' local data to the central server where content popularities are predicted for future content offloading. These schemes result in higher communication exchange overheads due to large data exchange between entities, and also prone to users' data privacy issues [10].

In order to tackle the above issues several distributed caching schemes [11]–[18] have been proposed. These schemes follow federated learning paradigm in which each base station (BS) i.e., small cell trains own content prediction model based on the local data and transmit the trained model updates to the global model over underlying communication channel. Generally, in case of FRANs, fog server hosts the global model. On receiving the local model updates from the associated cells, the global model executes federated model weight aggregation and sent model updates back to the cells. These federated learning based schemes such as [11], [16] assumes that communication channel between cells and the fog server is perfect and there is no noise effect. However, under dense small cell networks, the channel error and poor state is more serious problem [19] due their lower antenna heights

Sanaullah Manzoor is affiliated with Information Technology University, Pakistan and University of Glasgow, United Kingdom.
Adnan Noor Mian is affiliated with Information Technology University, Pakistan. Corresponding Email: sanaullah.mazoor@itu.edu.pk

ISBN 978-3-903176-37-9 © 2021 IFIP

which leads to more service blockages, channel fadings, and more frequent interruptions. Under such scenario, the global model averaging will result in aggregation of channel errors into the model weights. The erroneous model updates if shared to the local cells may compromise the caching decisions. Consequently, such noisy federated model will results in lower cache hits and may leads to downloading of large volumes of unwanted data at network edge which will consume extra bandwidth and as a results there will be poor quality of service.

In order to mitigate the effect of aggregated model uncertainty due to communication errors in unreliable channel, we proposed a robust federated learning based content caching (RFC) framework as shown in Figure 1. To authors' best knowledge this is the first study that incorporates channel effects during federated model aggregations. Our framework formulates a deep neural network-based federated future content score prediction model that learns contextual information between end user and its requested previous files while restricting data locally in order to predict future content score. The RFC framework shares the learned weights from small cells (SCs) to fog sever for global model aggregation. Fog server exploits these local updates to train generative adversarial network (GAN) which performs min-max optimization to discriminate between actual and erroneous local model updates before global model aggregation. This GAN model introduces robustness against channel errors for model communication exchange. Extensive simulations have been carried out validate the performance of the proposed RFC framework. The major contribution of this work as follows:

- We proposed a robust federated learning based content caching (RFC) framework that minimizes the effect errors in the federated model aggregation under an unreliable channel.
- Our RFC framework develops a deep neural network-based federated future content score prediction model that learns contextual information between end user and its requested previous files while restricting data locally in order to predict future content score. The RFC framework shares the learned weights from small cells (SCs) to fog sever for global model aggregation.
- We trained a generative adversarial network (GAN) which performs min-max optimization to discriminate between actual and erroneous local model updates before global model aggregation.
- Extensive simulations carried out validate the performance of the proposed RFC framework against the state of the art federated learning based caching schemes.

The rest of paper is organized as follows: section II provides research gap and literature review. Section III formulates problem statement. Section IV explain the proposed RFC framework. Section V contains simulation setup and, results and discussion. Finally, section VI concludes the paper.

II. Related Work

Owing to caching storage constraint, users' future content prediction has gained a lot of attention which aims to fetch proactively the most likely to be asked contents at the local caches. In last decade, several edge caching schemes have been proposed [6]–[9], [11]–[18], [20]. These schemes can be classified based on their mode of operation and underlying performance objective. Caching decision mechanisms are either centralized or distributed based on mode of operation. Centralized mechanisms such as [13], [15], [16] train caching model centrally which have scalability and data privacy issue as all the data firstly have to send to central sever for model training and then model decision is shared with the associated cells.

In order to alleviate the scalability and data privacy issues of centralized learning, federated learning is considered as an alternative approach. Therefore, to-date various federated learning based schemes have been introduced. Here, we discuss only closely related works in the future content caching. Wang et al. proposed a federated learning based cooperative content caching for internet of things networks [21]. Some works on compressed model updates have been reported such as Cui et al. [22] introduced a block-chain based compressed federated caching model in which they share compressed model with clients. Similarly, Xiao et al. proposed federated learning-based content offloading scheme for F-RANs [23]. In an other work, Xue et al. proposed federated learning based edge caching for E-health system [24].

Precisely, all these works ignored the channel conditions between cells and fog server during federated weight averaging and reported results under ideal condition. This assumption practically is not viable as pico and femto cell based networks may face more signal disruptions, irregular load distributions, and often blockage due to short height as compared to marco cells. Thus, for federated learning based scheme cannot neglect effects of channel noise which may impact federated averaging of local models. Therefore, in this work we considered this issue and proposed robust federated learning based future content prediction scheme that mitigates the effects of channel noise during federated averaging and share noise effect free global model updates with local models.

III. System Model

In this section, we describe our system model and the proposed approach. In our system model, a fog sever is supervising B dense small cells $b = \{1, 2, 3, ..., B\}$. System has U users i.e., $u = \{1, 2, ..., U\}$. Figure 1 show the system model. Fog sever is connected to core network functionalities including mobility management entity (MME), and serving gateway (S-GW). Furthermore, we define these entities: let $\mathbf{g} = g_1, g_2, ..., g_B \in \{0, \mathbb{Z}^+\}$ represent the backhaul link capacities between cells and the fog server, where $\mathbb{Z}$ is system bandwidth. Now, let $\mathbf{f} = [f_1, f_2, ..., f_F]$ shows file library in which each file is atomic and has a size v_i, and file size vector is given as: $\mathbf{v} = [v_1, v_2, ..., v_F]$.

The signal to interference and noise ratio (SINR) η_{bu} between cell b and associated UE u is given as followed:

$$\eta_{bu} = \frac{P_b \mathcal{H}_{bu}}{\sum_{j \in B/\{b\}} P_j \mathcal{H}_{ju} + \gamma^2} \tag{1}$$

where $\mathcal{P}_b$, $\mathcal{H}_{bu}$, and γ^2 is cell transmission power, channel gain, and additive white Gaussian noise (AWGN) respectively.

ISBN 978-3-903176-37-9 © 2021 IFIP

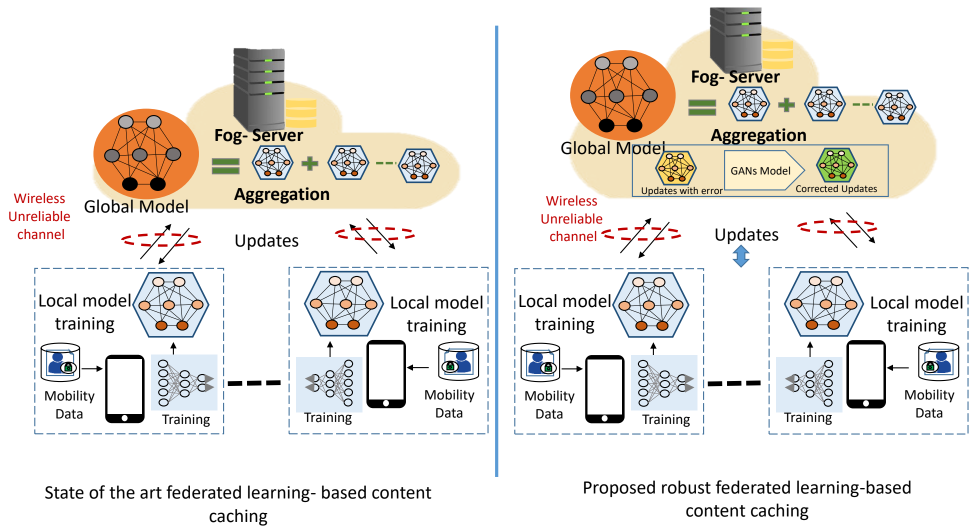

Fig. 1. Proposed federated learning based robust caching framework

Now the maximum donwlink (DL) achievable rate from cell b to UE u given P_b is represented as:

$$R_{bu} = \mathcal{S}log_2\left(1 + \frac{P_b\mathcal{H}_{bu}}{\sum_{j\in\mathcal{B}/\{b\}} P_j\mathcal{H}_{ju} + \gamma^2}\right) \quad (2)$$

where $\mathcal{S}$ is bandwidth of each physical resource block (PRB) which is typically $180kHz$ in orthogonal frequency division multiple access (OFDMA) based cellular networks.

IV. Proposed Robust Federated Learning Based Future Content Prediction Model

This section explains the federated learning based users' future content prediction model. Each cell hosts users' file demand matrix $\mathbf{D} \in \mathbb{R}^{U\times F}$ which keeps the records of various content F asked by the users U. Each entry $d_{i,j}$ in the matrix $\mathbf{D}$ is file rating score from i_{th} user against the file j_{th}. If score doesn't exist then corresponding entry $d_{i,j}$ will be zero. In matrix $\mathbf{D}$ the most requested files will be highly rated while the least requested files will be lowest rated or zero. We can represent the matrix $\mathbf{D}$ as:

$$\mathbf{D} = \begin{bmatrix} D_1 \\ D_2 \\ \vdots \\ D_U \end{bmatrix} = \begin{bmatrix} d_{1,1} & \dots & d_{1,F} \\ d_{2,1} & \dots & d_{2,F} \\ \vdots & \vdots & \vdots \\ d_{U,1} & \dots & d_{U,F} \end{bmatrix} \in \{0, \mathbb{Z}^+\}^{U\times F} \quad (3)$$

In matrix $\mathbf{D}$ we will have $F \gg U$ because each user can not request every file from the file library $\mathbf{f}$. Thus, network can rate only demanded (requested) files. Eventually, the $\mathbf{D}$ will be a spare matrix will many 0 enteries. In order to download contents proactively, we need to determine full content score/rating matrix $\mathbf{D}^{'}$.

In our proposed approach each cell formulates the future content rating score prediction problem as a supervised learning problem and exploits a deep neural network (DNN) model to learn the contextual information between users and files. Here, is details of the proposed DNN model: let user's latent feature matrix $\mathbf{M} \in \mathbb{R}^{U\times a}$ in which the vector $\mathbf{M}_i$ represents i_{th} users' features and a shows dimensions of the vector $\mathbf{M}_i$. Similarly, files' latent feature matrix $\mathbf{N} \in \mathbb{R}^{F\times k}$ with feature vector $\mathbf{N}_i$ and k represents dimensions of the vector $\mathbf{N}_i$. For model feature learning, we encoded the vector $\mathbf{M}$ and $\mathbf{N}$ given as follows:

$$M_i = NN\left(OneHot(i)\right) \quad (4)$$

$$N_j = NN\left(OneHot(j)\right) \quad (5)$$

where $oneHot(i)$ represents one-hot encoding vector of $\mathbf{M}_i$ and term $NN(x)$ denotes the output of DNN model. Figure 2 the proposed DNN based model architecture which takes user and files' latent feature as input and predicts future file rating score. In the proposed DNN based content prediction model, the input vector $\mathbf{x}_O$ is formulated using concatenation of the latent feature vector and is given as:

$$x_0 = concatenate\left(\mathbf{M}_i, \mathbf{N}_i\right) \quad (6)$$

where *concatenate()* function joins $\mathbf{M}$ and $\mathbf{N}$ vectors and the its output is propagated to next hidden layer which is given as:

$$x_1 = ReLU\left(\mathbf{W_1}x_o + v_o\right) \quad (7)$$

where $\mathbf{W_1}$ and v_o denotes the weight matrix been the input and first hidden layer, and bias vector respectively. In order to introduce non-linearity in our DNN model we used *ReLU()* activation function. At final layer of DNN model we used *softmax()* classification function to predict the rating terms of matrix $\mathbf{D}^{'}$, i.e., $d^{'}_{i,j}$ and is given as:

$$Y^{'} = softmax\left(\mathbf{W_o}x_h + v_{out}\right) \quad (8)$$

where x_h, $\mathbf{W_o}$, and v_{out} represent output of hidden layer, weight matrix, and bias vector respectively.

Each cell host local DNN content future score prediction model of configuration: at input layer users' latent vector $\mathbf{M}_i$ and files' latent feature vector $\mathbf{N}_i$ are fed into model after concatenation. Model contains five hidden layers with these respective number of neurons in each layer 200, 100, 50, 20, 10. Output layer contain *softmax()* probabilities vector of predicted users' future content rating score $d^{'}_{i,j}$. Each local

ISBN 978-3-903176-37-9 © 2021 IFIP

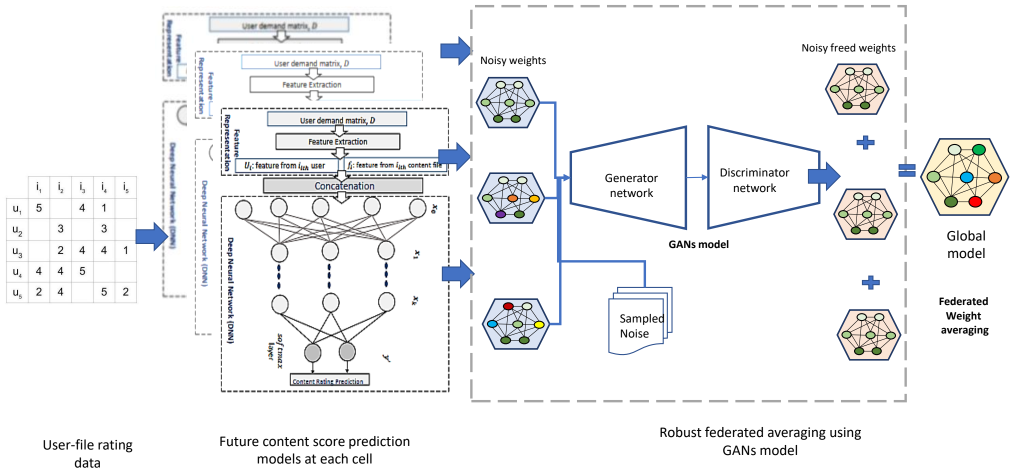

Fig. 2. Working mechanism of robust federated learning based future content predictions

model tries to minimize categorical cross-entropy loss which is given as:

$$\psi = -\sum_{k=1}^{K} y_k log(y_k^{'}) \tag{9}$$

where K represents the total number of classes in the score rating matrix. Each local model perform training on the local data up-to specified number of Epochs and forwards its weights $\mathbf{W}_{Lb}$ to fog sever, where federated weight averaging is performed. The global model hosted at fog server gets all local model weights and performs federated weight averaging which is given as:

$$\mathbf{W}_G = \mathbf{W}_{L1} + \mathbf{W}_{L2} + ... + \mathbf{W}_{LB} \tag{10}$$

where $\mathbf{W}_G$ and $\mathbf{W}_{Lb}$ represent weights of global model and weights of local model respectively. As each local model weight $\mathbf{W}_{Lb}$ is transmitted to fog server via noisy channel then there will be notable amount of noisy in each weight matrix which is expressed as:

$$\mathbf{W}_{Lb}^{'} = \mathbf{W}_{Lb} + \beta \tag{11}$$

where β represents the noise in the aggregated federated weight averaging matrix due to poor communication channel. The term β will impact on actual learned weight matrix $\mathbf{W}_{Lb}$ and may leads to longer convergence times. Thus, there is need to mitigate this noise impact at fog server. In order to tackle this issue we proposed a generative adversarial neural networks (GANs) based noise mitigation model that learns actual distribution of $\mathbf{W}_{Lb}$ weight matrix and results in the noise-free weight matrix. Figure 2 shows the proposed GANs model. The GANs model has a generator **G** and discriminator **D** network. The generator model takes noise β and tries to generate samples of actual local model weight matrix $\mathbf{W}_{Lb}^{'}$. The discriminator network has information of actual $\mathbf{W}_{Lb}$ weight matrix, then it guides generator to produce as close samples nearly same to $\mathbf{W}_L$ matrix. Thus GANs model learns actual distribution of $\mathbf{W}_{Lb}$ matrix and able to distinguish between noisy wight matrix $\mathbf{W}_{Lb}^{'}$ and actual $\mathbf{W}_L$ matrix. In other words, GANs model plays **minmax** optimization in order to learn true distribution of actual data, and is given as:

$$min \quad max \quad V(G,D) = \mathbb{E}_{x\sim p_{data}}[log(D(x)]+$$
$$\mathrm{E}_{\beta\sim p_\beta}[log(1-(D(G(\beta)))]$$

where output of discriminator **D** reveals that the predicted sample belongs to actual data or noisy data, thus discriminator **D** wants to maximize classification difference between actual and noisy data. Whereas generator **G** minimizes the learning error of above objective function by producing very much similar data to actual data distributions. Fog sever passes each learned federated averaged weight matrix $\mathbf{W}_{Lb}^{'}$ and gets noise free $\mathbf{W}_{L_b}^{'}$ matrix which is very much near to actual weight matrix $\mathbf{W}_{Lb}$. This is how our proposed robust federated scheme allows fog sever to mitigate the impact of noise during federated weight averaging Eq. 10 procedure.

Here we define the training procedure of our GANs model: model weights are taken as image data and feed into the GAN model. Further, we trained generator with two dense layers and with 15 latent dimensions. Generator network has *ReLU()* activation function. The discriminator network contains two layers of dense network with 25 latent dimensions and binary cross-entropy loss function is used for the training. During the training of GANs model we provide it the actual $\mathbf{W}_{Lb}$ weight matrix when there is no noise in the channel and the noisy $\mathbf{W}_{Lb}^{'}$ weight matrix. Under our small cell based system model the channel noise between cell and fog server is sampled from Weibull distribution [25]. The GANs model is trained for 1000 epochs with batch size of 16 samples and learning rate of 0.003.

Now on receiving corrected model weight updates $\mathbf{W}_{L_b}^{'}$ from fog sever, each local DNN content future score prediction model start next communication round $t+1$ and trains its model with specified number of epochs. After robust federated weight averaging each local model has overall network users' and files' context, thus it is able to rate users' future contents based on the learned model. Now, each local DNN model predicts i_{th} user's future rating score $d_{i,j}^{'}$ for the j_{jth} file and is given as:

$$d_{i,j}^{'} = \mathbf{argmax}\,[\hat{y_1}, \hat{y_2}, ..., \hat{y_K}], \forall k \in K \tag{12}$$

ISBN 978-3-903176-37-9 © 2021 IFIP

where K is the total number if classes in the future content score rating matrix $\mathbf{D}^{'}$. Algorithm 1 explains the implementation details of the proposed robust federated learning based content score prediction model.

Algorithm 1: Implementation of Robust Federated Learning based Content Caching

Get user feature representation vector M_i
Get content feature representation vector N_i
Perform concatenation of M_i and N_i vectors using Eq. 6
Initialization of learning rate α, local epochs E, and data batch size ζ
Procedure GlobalUpdates(W_c, U,)
$\mathbf{W_c} \leftarrow \mathbf{W_c^{'}} + \Delta_k$
return $\mathbf{W_c}$ trained local weights
for $t \in T$ **do**
  for $(b \in B)$ **do**
    If Download global model weights if exist else initialize weights
    else
    Train local DNN model as described in section IV
    Invoke GAN model for model noise mitigation
    Perform global federated model averaging Send global model updates to associated cells
  Get future file rating score $d_{i,j}^{'}$
terminate

V. Performance Evaluation and Results

This section describes performance evaluation of the proposed robust federated based content caching. We assesses the performance of caching model in terms of error loss and prediction accuracy. The performance is also assessed by varying the number of cells and observed system performance.

A. Simulation Setup

The proposed scheme is evaluated using real-world sparse public data *ML100K* provided by *MovieLens* [26]. The dataset contains $100,000$ rating values between $[1,5]$ from 943 unique users. The number of unique files are 1682. In this data, each user has rated at least 20 files. We used 70% of data form model training whereas 30% data is used for caching model assessment. The simulations are carried out *Google Colab* environment [27]. For noise generation we used Weibull distribution with $a = 3.5$. For GAN model training we feed network weights as an image and also provided images with noise.

B. Results and Discussion

In this section we cover simulation results and discussions. We evaluated system performance under perfect channel based federated learning, and noisy channel based federated learning. Figure 3 shows model loss under conditions (i) perfect channel conditions, (ii) noisy channel, and (iii) noisy channel model with GAN-aided FL model along with communication rounds. It can observe that lowest loss is under perfect channel conditions and highest under noisy channel. The lowest loss under perfect channel condition indicate that model is learning

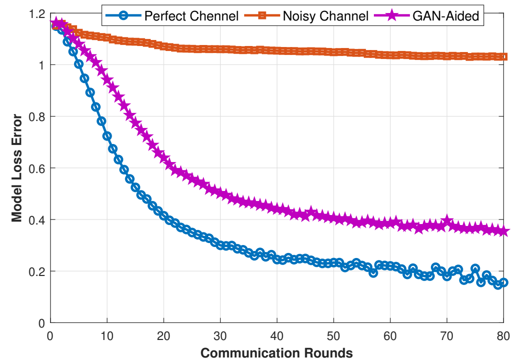

Fig. 3. Model loss of federated content caching model (i) perfect channel conditions, (ii) noisy channel ,and (iii) noisy channel model with GAN-aided FL model

very fine under federated learning and highest loss under under noisy channel reveals that noise is impacting learning process and thus, model is not converging. However, when we incorporate the GAN model prior to federated weight averaging, model performance is improved and loss is getting lower which also indicates that model is converging.

Further, we assessed system performance under future content prediction model accuracy. Figure 4 shows the prediction accuracy under perfect channel conditions, model with noisy channel, and noisy channel model with GAN-aided FL model along with communication rounds. We can see that under first can when there is no communication noise during federated model averaging, accuracy is higher at the end of communication rounds. This accuracy drops to 55% when there is communication noise during federated model averaging. However, GAN-aided model is not much effected with noise as it can learn the actual model weight model distributions and results better accuracy.

Furthermore, we extended the experimentation for number of cell B in the system and Figure 5 shows model prediction accuracy along with communication rounds. Under perfect channel conditions, model has the accuracy of $0.857, 0.852, 0.811$,and 0.805 when number of cells are varied from 2 to 5. Under noisy conditions without GAN model, accuracies are $0.5803, 0.535, 0.543$, and 0.533. The system accuracy improves in the presence of communication channel noise when we employee the proposed GAN model and here are the accuracy results, $0.804, 0.776, 0.775$, and 0.761 respectively which highlights the generalized performance of the proposed scheme. Overall, GAN-aided federated model yields 77.9% prediction accuracy while same federated model under noisy channel give only 54.8% accuracy.

VI. Conclusion

In this paper, we introduced a novel robust federated learning-based content caching scheme for F-RANs that mitigates the effect of communication channel noise. In our proposed robust FL approach, each cell employs a deep neural network (DNN)-based model to predict users' future files rating score based on user and file contextual information and shares its learned weights to the fog server. Fog server is responsible for global weight averaging. Prior to FL weight

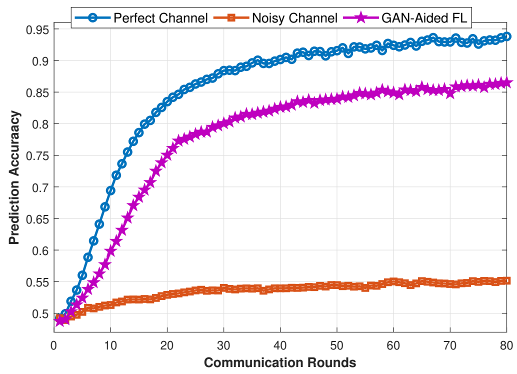

Fig. 4. Prediction Accuracy of federated content caching model (i) perfect channel conditions, (ii) noisy channel and (iii) noisy channel model with GAN-aided FL model

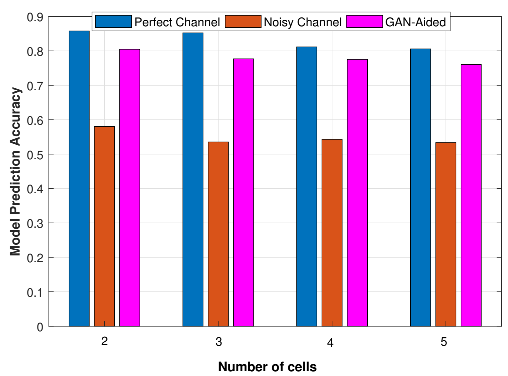

Fig. 5. Model accuracies under varying the number of cells for systems (i) perfect channel conditions, (ii) noisy channel and (iii)GAN-aided FL model

averaging fog sever feed incoming local model weights to a generative adversarial neural network (GANs) model which differentiates between noisy and actual federated weights, and passes only actual weights based on the distribution of the weight matrices. We performed extensive simulations to validate the performance of our proposed FL-based caching approach. Results show that the GAN-aided federated model yields 23.1% more prediction accuracy as compared to the federated noisy model without GANs based noise mitigation. In the future, we aim to extend experimentation for channel noise variations and estimation for noise tolerance levels for federated weight averaging.

References

[1] W. P. Cisco, “Cisco, visual networking index: Global mobile data traffic forecast update, 2013–2018 white paper,” *Document ID*, vol. VI, USA, Mar, 2017.

[2] S. Ray, Y. Jin, and A. Raychowdhury, “The changing computing paradigm with internet of things: A tutorial introduction,” *IEEE Design & Test*, vol. 33, no. 2, pp. 76–96, 2016.

[3] W. Paper, “Cisco VNI forecast and methodology, 2015-2020,” *Document ID*, vol. VI, USA, Jun, 2017.

[4] L. Li, G. Zhao, and R. S. Blum, “A survey of caching techniques in cellular networks: Research issues and challenges in content placement and delivery strategies,” *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 1710–1732, 2018.

[5] Y. Shi and Q. Ling, “An adaptive popularity tracking algorithm for dynamic content caching for radio access networks,” in *2017 36th Chinese Control Conference (CCC)*. IEEE, 2017, pp. 5690–5694.

[6] A. Sadeghi, F. Sheikholeslami, and G. B. Giannakis, “Optimal and scalable caching for 5g using reinforcement learning of space-time popularities,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 180–190, 2017.

[7] P. Yang, N. Zhang, S. Zhang, L. Yu, J. Zhang, and X. Shen, “Content popularity prediction towards location-aware mobile edge caching,” *IEEE Transactions on Multimedia*, vol. 21, no. 4, pp. 915–929, 2018.

[8] S. M. Azimi, O. Simeone, A. Sengupta, and R. Tandon, “Online edge caching and wireless delivery in fog-aided networks with dynamic content popularity,” *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 6, pp. 1189–1202, 2018.

[9] Z. Zhang and M. Tao, “Deep learning for wireless coded caching with unknown and time-variant content popularity,” *IEEE Transactions on Wireless Communications*, 2020.

[10] Z. Zhao, C. Feng, H. H. Yang, and X. Luo, “Federated-learning-enabled intelligent fog radio access networks: Fundamental theory, key techniques, and future trends,” *IEEE Wireless Communications*, vol. 27, no. 2, pp. 22–28, 2020.

[11] Z. Yu, J. Hu, G. Min, H. Lu, Z. Zhao, H. Wang, and N. Georgalas, “Federated learning based proactive content caching in edge computing,” in *2018 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2018, pp. 1–6.

[12] H. Feng, Y. Jiang, D. Niyato, F.-C. Zheng, and X. You, “Content popularity prediction via deep learning in cache-enabled fog radio access networks,” in *2019 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2019, pp. 1–6.

[13] F. Jiang, Z. Yuan, C. Sun, and J. Wang, “Deep q-learning-based content caching with update strategy for fog radio access networks,” *IEEE Access*, vol. 7, pp. 97 505–97 514, 2019.

[14] Y. Jiang, X. Cui, M. Bennis, F.-C. Zheng, B. Fan, and X. You, “Cooperative caching in fog radio access networks: a graph-based approach,” *IET Communications*, vol. 13, no. 20, pp. 3519–3528, 2019.

[15] Y. Sun, M. Peng, and S. Mao, “A game-theoretic approach to cache and radio resource management in fog radio access networks,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 10, pp. 10 145–10 159, 2019.

[16] S. Yan, M. Jiao, Y. Zhou, M. Peng, and M. Daneshmand, “Machine learning approach for user association and content placement in fog radio access networks,” *IEEE Internet of Things Journal*, 2020.

[17] G. S. Rahman, M. Peng, S. Yan, and T. Dang, “Learning based joint cache and power allocation in fog radio access networks,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 4, pp. 4401–4411, 2020.

[18] L. Hu, Y. Miao, J. Yang, A. Ghoneim, M. S. Hossain, and M. Alrashoud, “If-rans: Intelligent traffic prediction and cognitive caching toward fog-computing-based radio access networks,” *IEEE Wireless Communications*, vol. 27, no. 2, pp. 29–35, 2020.

[19] Y. Park, J. Heo, and D. Hong, “Spectral efficiency analysis of ultra-dense small cell networks with heterogeneous channel estimation capabilities,” *IEEE Communications Letters*, vol. 21, no. 8, pp. 1839–1842, 2017.

[20] S. Manzoor, S. Mazhar, A. Asghar, A. Noor Mian, A. Imran, and J. Crowcroft, “Leveraging mobility and content caching for proactive load balancing in heterogeneous cellular networks,” *Transactions on Emerging Telecommunications Technologies*, p. e3739, 2019.

[21] X. Wang, C. Wang, X. Li, V. C. Leung, and T. Taleb, “Federated deep reinforcement learning for internet of things with decentralized cooperative edge caching,” *IEEE Internet of Things Journal*, 2020.

[22] L. Cui, X. Su, Z. Ming, Z. Chen, S. Yang, Y. Zhou, and W. Xiao, “Creat: Blockchain-assisted compression algorithm of federated learning for content caching in edge computing,” *IEEE Internet of Things Journal*, 2020.

[23] T. Xiao, T. Cui, S. Islam, and Q. Chen, “Joint content placement and storage allocation based on federated learning in f-rans,” *Sensors*, vol. 21, no. 1, p. 215, 2021.

[24] X. Xiong, K. Zheng, L. Lei, and L. Hou, “Resource allocation based on deep reinforcement learning in IoT edge computing,” *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 6, pp. 1133–1146, 2020.

[25] X. Li, J. Li, and L. Li, “Performance analysis of impaired swipt noma relaying networks over imperfect weibull channels,” *IEEE Systems Journal*, vol. 14, no. 1, pp. 669–672, 2019.

[26] http://grouplens.org/datasets/movielens/, “Movielens Dataset,” March 2018.

[27] E. Bisong, “Google colaboratory,” in *Building Machine Learning and Deep Learning Models on Google Cloud Platform*. Springer, 2019, pp. 59–64.

ISBN 978-3-903176-37-9 © 2021 IFIP