AI-BASED NETWORK TOPOLOGY OPTIMIZATION SYSTEM

Han Zengfu, Kong Jiankun, Wang Zhiguo, Zhang Yiwei, Liu Ke, Pan Liang, Li Sicong, Wu Desheng

China Mobile Shandong Branch, 20569, Jingshi road, Jinan, Shandong

NOTE: Corresponding author: Han Zengfu, hanzengfu@sd.chinamobile.com

Abstract – Existing network topology planning does not fully consider the increasing network traffic and problem of uneven link capacity utilization, resulting in lower resource utilization and unnecessary investments in network construction. The AI-based network topology optimization system introduced in this paper builds a Long Short-Term Memory (LSTM) model for time series traffic forecasting, which uses NetworkX, a Python library, for graph analysis, dynamically optimizes the network topology by edge deletion or addition based on traffic over nodes, and ensures network load balancing when node traffic increases, mainly introducing the LSTM forecasting model building process, parameter optimization strategy, and network topology optimization in some detail. As it effectively enhances resource utilization, this system is vital to the optimization of complex network topology. The end of this paper looks forward to the future development of artificial intelligence, and suggests the possibility of how to cooperate with operator networks and how to establish cross-border ecological development.

Keywords –Artificial intelligence, capacity utilization, communication network, traffic forecast, network topology

1. INTRODUCTION

With the development of 5G technology, operators need to rebuild or expand their transport networks, as their existing network topology planning does not fully consider the increasing network traffic and problem of uneven link capacity utilization [1]. The utilization of more than 40% of the transmission links of the existing networks is low [2,3], which increases operators' network construction costs. Therefore, how to optimize the network structure to make it adapt to future network changes and how to improve resource utilization to save construction costs have greatly challenged today's operators [7].

For complex networks, topology analysis requires exponential computing [4,5], making manual network topology optimization very difficult. With the development of Artificial Intelligence (AI) technology, Machine Learning (ML) algorithms offer a helping hand to resolve complex issues [6,7]. A new ecosystem which needs to be established to advance such technical developments as applying AI and big data technologies to networks is new to the industry, and the "AI-ML in 5G Challenge" organized by ITU paves the way for such study. This paper introduces an AI-based network topology optimization system that makes in-depth analysis of network topology and proposes a solution to maintain high resource utilization when network traffic keeps growing [2]. It is especially crucial for the optimization of complex network topology.
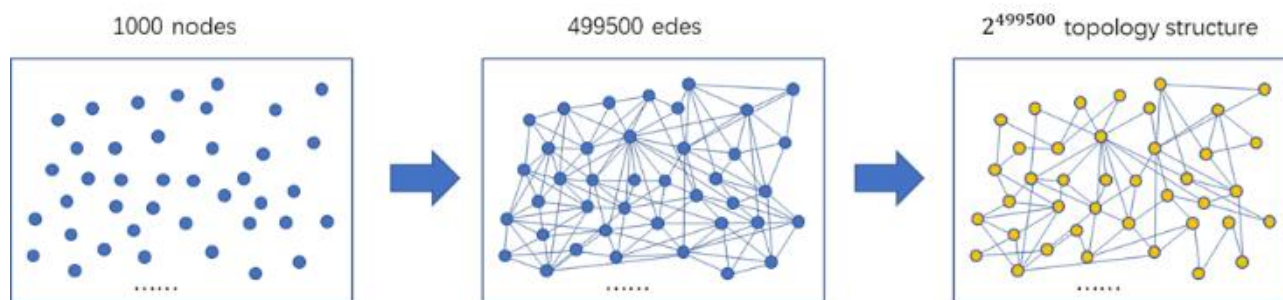

Fig. 1 – Complicated network topology analysis

2. LSTM MODEL FOR TRAFFIC FORECASTING

Network traffic forecasting is made based on network elements and affected by various factors such as the location of network elements, weather, and traffic on different base stations. Forecasting is mainly implemented via Gompertz models or time series forecasting models like LSTM and Prophet models [8,9,10]. The traffic forecast made based on traditional Extended Gompertz Models (EGMs) cannot reflect traffic differences between working days and holidays. However, the traffic predicted based on time series forecasting models with deep learning is more accurate, as history data can be learned and used during the traffic forecast [11].

2.1 Traffic forecasting model with deep learning

The short-term traffic forecast predicts the data of the next few days. As daily traffic features strong periodicity, traffic curves of the days in weeks with similar attributes are almost the same. In view of that, this paper proposes an LSTM model [10,11], a recurrent neural network architecture that gives full play to the correlation of traffic at different times, for traffic forecasting. The model is trained with time-stamped traffic (for example, traffic of working days or holidays), and outputs more accurate predicted data through iteration technologies [12].

2.2 Establishment of the LSTM model

In the study, the LSTM model was built with TensorFlow2.0 and traffic per hour was taken as a sample. The traffic samples of the last 20 days were processed first, and the traffic over each network element was listed by time series after processing [13,14]. The following four steps describe how the LSTM model was built and traffic forecasting was conducted.

1) Make data sets: The traffic samples of the first 15 days were used for training and testing, and those of the remaining 5 days were used for result verification. The training data set included 75% traffic samples of the first 15 days, and the testing data set contained the other 25%. The initial default time sliding window was 24.

2) Establish the LSTM model: Add two LSTM layers to the LSTM model for traffic forecasting.

3) Train the model: Input training data and test data into the LSTM model, and set epochs and batch_size to 50 and 32 respectively.

4) Conduct traffic forecasting: Forecast the traffic of the next hour through iteration technologies based on the time sliding window. Enhance the forecasting efficiency via multiprocessing technologies and concurrent processes.

2.3 Optimization of the LSTM model

At the beginning of the study, the traffic forecasted with the default parameters of the model was quite different from the data (traffic of the last five days) which remained for result verification. After repeated comparison and analysis, the most proper epochs, batch_size, and slide_window were determined. The following shows the way that we used in the study to determine those parameters.

First, set the batch_size and slide_window to fixed values, and the epochs to 50, 100, 200 and 400 respectively. Then, evaluate the impact of each epoch's value on the errors in traffic forecasting in terms of the Root Mean Square Error (RMSE) and running time. The results indicate that "epochs=200" is the optimal choice.

After the model was trained for 50, 100, 200, and 400 times respectively, the RMSE generated accordingly, predicted traffic volume, and the time that training costs were recorded in the following table.

Table 1 – Optimization of the model's epochs

Times	RMSE	Predicted Traffic Volume (GB)	Running Time(s)
50	16.246	251.3	55
100	3.714	279.9	85
200	1.9	292.4	148
400	2.3	293	305

As Table 1 shows, when the model was trained for 200 times, it generated the smallest RMSE and cost the shortest time. However, when the training was conducted for 50, 100, and 400 times, both the RMSEs that were generated accordingly, and the running time of each training failed to meet the requirements. Therefore, the model was finally trained for 200 times to ensure both high efficiency and accuracy.

When epochs were set to 200 and slide_window to a fixed value, the result of repeated comparison of the errors caused by configuring different batch_size showed that batch_size=64 is the best choice.

When epochs=200 and batch_size=64, "slide_window=48" causes the minimum error in traffic forecast.

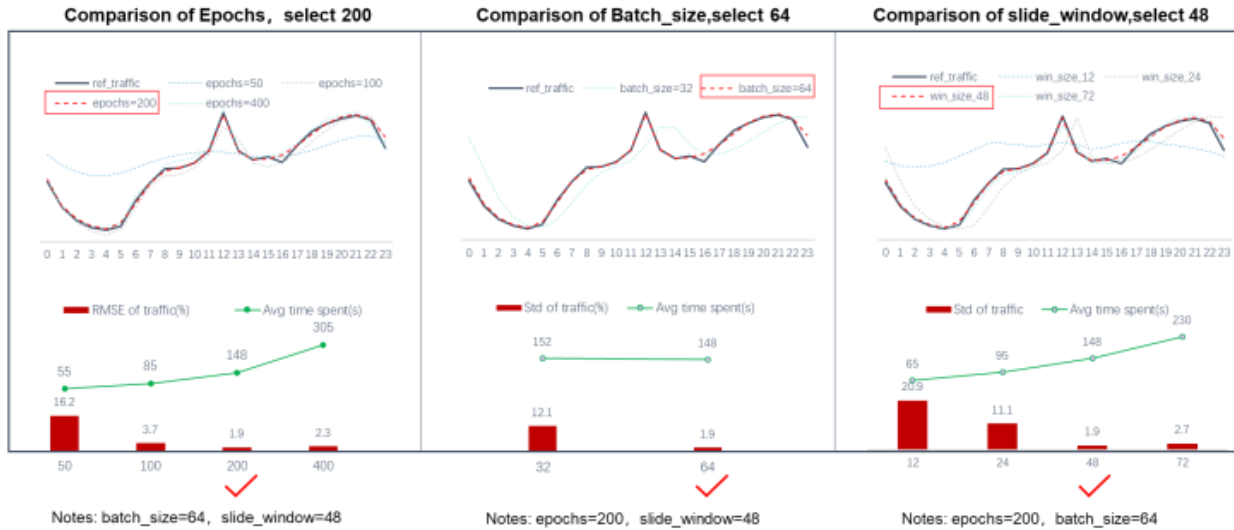Fig. 2 shows the errors in traffic forecast when the model is configured with different parameter values.

Fig. 2 – Model set with different parameter value

In view of accuracy of the forecast and running efficiency, the parameter values of the traffic forecasting model were selected through multiple rounds of testing. The final values used in the system ensure that the average error in traffic forecast is under 3%, as shown in Fig. 3 below.
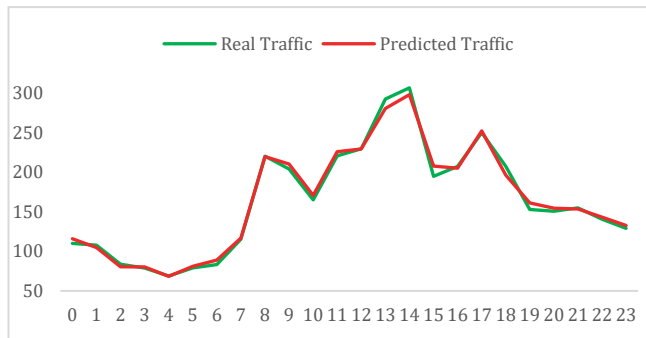

Fig. 3 – Comparison between predicted data and the data for result verification

After optimization, we input the same sample into the new LSTM model (which was named Our-LSTM) and other five well-known models respectively, namely ARIMA, LightGBM, Prophet, LSTM, and DeepAR, and made some comparisons. For specific information, refer to Table 2.

Table 2 - Comparisons of traffic forecasting models

Model	Absolute Accuracy (%)	Relative Accuracy
ARIMA	18.36	2.6721
LightGBM	20.31	1.8742
Our-LSTM	3.01	0.6552
Prophet	8.88	2.3516
LSTM	15.02	1.7471
DeepAR	16.3	1.6913

As Table 2 shows, both the absolute and relative accuracy achieved through Our-LSTM are better than those achieved based on the initial LSTM, proving that the hyper-parameters of Our-LSTM after optimization are more suitable for the network. When compared with other time series forecasting models listed in the above table, Our-LSTM with the highest absolute accuracy and relative accuracy is superior to them in short-term time series forecasting.

3. NETWORK TOPOLOGY ANALYSIS FOR OPTIMIZATION

Topology optimization is quite complicated. Existing network topology planning can hardly satisfy future traffic increase or solve the problem of uneven load balancing, as it is made based on the current network load. Our research is to resolve this problem through dynamic network topology optimization. Specifically, the topology optimization system introduced in this paper enables network load balancing and makes the topology satisfy future traffic growth.

NetworkX, a Python library for studying graphs and networks, is an effective tool for analyzing network topology, building network models, designing new network algorithms, and plotting network graphs. In this system, NetworkX is used to plot the network graph [15].

3.1 Network topology reshaping

3.1.1 Define nodes and links

As per different functions, significance, and capacity, the nodes in the network topology to be optimized can be classified into three types, namely, nodes G, H, J [16,17,18]. There are also three types of links, namely, main links, sub-links, and hanging links.

Link: A link in the network topology can be made of one main link, zero or multiple sub-links, and zero or multiple hanging links. The total number of nodes on each link does not exceed 30.

Main link: Each main link has the following features: 1) Its end nodes are node G/H. 2) Its intermediate nodes are node H or J. 3) The capacity of each intermediate node is the same. 4) The total number of nodes on one main link equals to or is less than 15.

Sub-link: Each sub-link complies with the following requirements: 1) Its end nodes are node G/H/J on main links (When the types of the start node and end node are different, they can only be node G/H), and intermediate nodes should be node J. 2) The capacity of a node J is smaller than or equal to that of either end node of the sub-link.

Hanging link: The link connects with a node G, or H, or J on one main link or sub-link through one edge only. The capacity of each node on a hanging link is smaller than or equal to that of the node where it connects, as shown in Fig. 4 below.
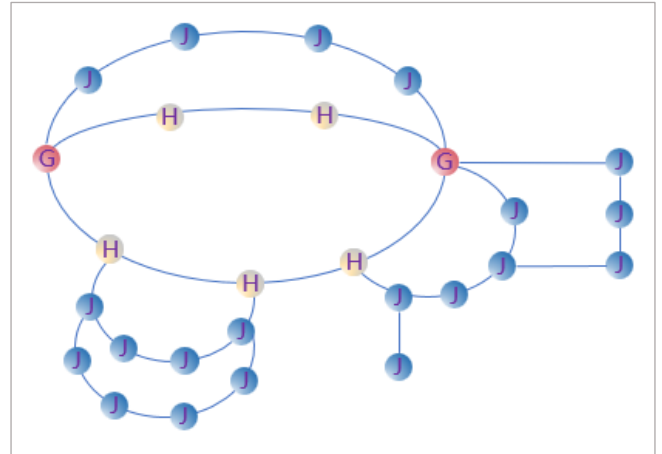
Fig. 4 – Structure of network topology

3.1.2 Introduction of neighbor links and neighbor nodes for faster topology analysis

Neighbor nodes and neighbor links in the network graph created with NetworkX can be defined [19]. As for neighbor nodes, one node on a specified link is adjacent not only to the appropriate node on the same link, but also to the nodes within a certain distance. Neighbor links refer to the links where two neighbor nodes (on different links) locate. The network topology in full status indicates that all neighbor nodes in this topology are connected. Such topology includes both existing links and all potential neighbor links. Network topology in optimized status refers to the network graph that we achieve by removing appropriate edges. Topology in full status and in optimized status can be mutually transformed.

Neighbor nodes and neighbor links become the only two factors that one needs to figure out when any node or link is to be analyzed, making iteration more simplified and topology analysis more efficient, as shown in Fig. 5 below.
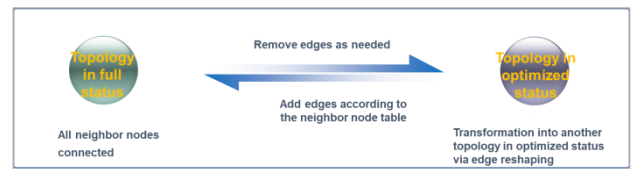
Fig. 5 – Full and optimized status

3.2 Innovative "Node-Removing Method" for highly efficient topology recovery

The network topology solution proposed in this paper is implemented based on edge reshaping. Specifically, links in the network topology need to be sorted out and saved in a link library.

First, find the main links whose end nodes are G or H via the DFS algorithm [20,21], and save all such main links in the link library. For cross-connected main links, select the one that is comparatively longer than the others in accordance with constraint rules on cross-connected links.

Regarding the sub-links and hanging links, the innovative "Node-Removing Method" is applied to sort them out when all main links have been identified. Specifically speaking, after main links in the network topology are removed, the related sub-links and hanging links will be disconnected accordingly. In this way, all the links in this network topology can be sorted out by category, and all the links of the topology will be recorded in the library by category.

1) Select one main link from the link library, and then remove it by using NetworkX.

2) Execute the connected_components() method to identify all connected sub-graphs [15].

3) Review all these sub-graphs and remove the sub-graphs that are not related to the current main link in accordance with the existing node connections.

4) Associate the sub-links and hanging links in the sub-graphs that have been removed in step 3 with the correct main link, and update the collection of neighbors' links in the link library.

5) Continue to remove other main links in the same way until all the sub-links and hanging links in the graph are associated with appropriate main links, as shown in Fig. 6 below.
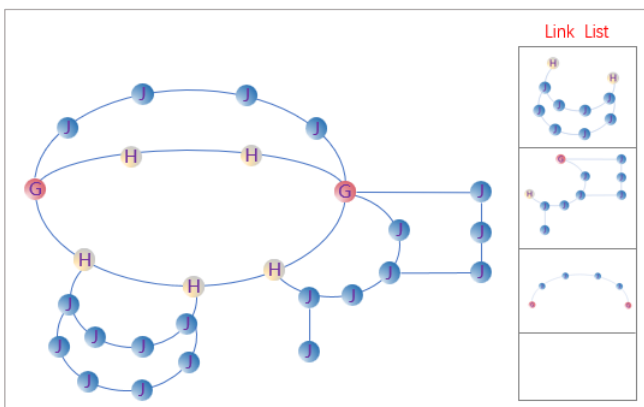

Fig. 6 – Node removing method

3.3 Evaluation of network topology

The following formula describes how the effect of network topology optimization is evaluated based on the network service rules, rules for assessing node levels [22,23], and the requirements for link load balancing.

$$\max \text{ object\_ratio}_{avg} - (E_{avg}+E_{min}+E_{max}) - \alpha \cdot \text{sub\_ratio} - \beta \cdot \text{hang\_ratio} \qquad (1)$$

Explain:

object_ratio$_{avg}$: Daily average of the ratio of links whose bandwidth utilization is optimized to the target range per hour

E_{avg}: Daily average of the E-value per hour that indicates network load balancing

E_{min}: Minimum E-value per hour of each day

E_{max}: Maximum E-value per hour of each day

sub_ratio: Proportion of nodes on sub-links to those on the entire link (daily average of values per hour)

hang_ratio: Proportion of nodes on hanging links to those on the entire links (daily average of values per hour)

α,β: Constant coefficients that are set to 0.02 and 0.05 respectively.

Edge addition: Distance between two nodes that are to be connected by adding an edge for topology optimization should be within 500 meters.

Number of nodes on each link: Maximally 30 nodes on each link.

Target range of link bandwidth utilization: $[\frac{3}{5}\mu, \frac{7}{5}\mu]$

Formula used to calculate E-value:

$$f_p = \frac{\sum_i^{nodes\ number} f_{xi}}{A} \qquad (2)$$

"E" is the variance of the link bandwidth utilization (fp) of all links in the network topology.

"μ" is the average of bandwidth utilization of all the links in the network topology. In the formula, fx is the traffic on the nodes (excluding the start and end nodes) on the link, and A stands for the maximum node capacity that one node (excluding the start and end nodes) on the link can possess.

3.4 Build and complete network topology optimization and find the optimal solution through iteration

Network topology optimization for load balancing can be implemented in three different ways [24,25,26,27], including link combination, partial link optimization, and the optimization by node transfer.

Identify links with heavy or low load based on their link utilization, check the link library to search for their neighbor links that carry low load or whose link utilization is within the target range, confirm whether any link combination can be conducted in accordance with the rules for link optimization, and combine the links that comply with the rules.

In the case where links cannot be optimized in the above way, partial link optimization can be implemented. 1) Identify the links adjacent to the sub-links or hanging links of the links with heavy load; 2) Opt for one appropriate neighbor link and connect those sub-links or hanging links with the neighbor link in accordance with the related rules.

The optimization by node transfer is applicable to all the links (including those whose link bandwidth utilization is within the target range). Specifically, the links are optimized through the transfer of some nodes with heavy load on these links to other links. It reduces the variance in the utilization of all links.

During network topology optimization, the iteration of network topology is carried out every hour, which is 24 times a day.

Regarding the topology optimization system, the network topology iteration carries out every hour after the first topology recovery. Therefore, each day, 25 scores on link 'status calculated' based on the evaluation formula can be achieved. During the topology iteration, the previous day's network topology with the highest score is used to start the next round of network topology optimization. Finally, the result of the optimization showed that the proportion of links with balanced load increased by 86%, as shown in Fig. 7 below.
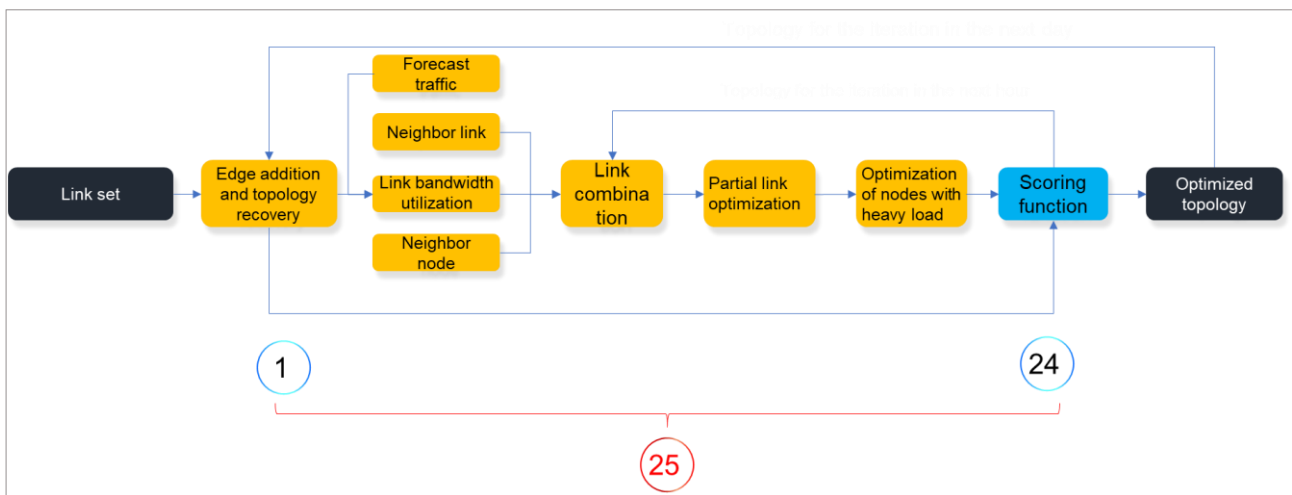

Fig. 7 – Network topology optimization process

3.5 Unique network topology restructuring for better topological structure

Network topology restructuring is to optimize the links that carry heavy or low load for a long time when basic network topology optimization is completed [28,29].

First, identify links with a heavy or low load for a long time. Second, split the link with heavy load into two links through either of the following two ways. One is to connect the link with two neighbor nodes (node G or H) respectively by adding two new edges. The other is to connect it with another main link that carries low load. By doing so, the iteration of the new topology carried out afterwards based on the new structure improves the utilization of all links [30].

The network topology restructuring aims to optimize network topology in a more complete way. In other words, it changes the status of the links that carry unbalanced load for a long time. In the study, we extended the distance between two neighbor nodes to 1000 meters and witnessed a 169% increase in the proportion of the links with a balanced load through the network topology restructuring [31,32], as shown in Fig. 8 below.
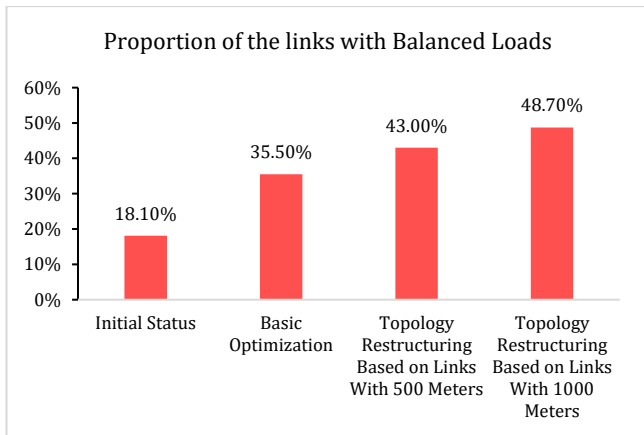
Fig. 8 – Proportion of links with balanced load in different optimization stages

The above-mentioned solution introduces a complete network topology optimization system for effective resolution of network topology problems, as shown in Fig. 9 below.
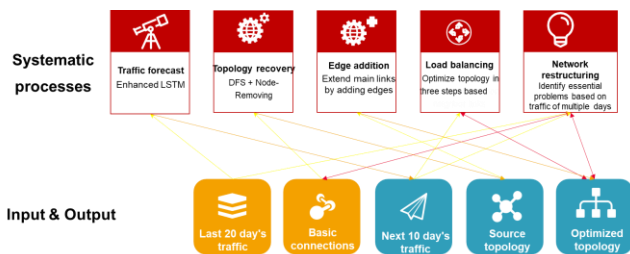

Fig. 9 – A complete network topology optimization system

4. CONCLUSION

Traffic forecasting and network load balancing are always under the spotlight of operators. During the study, all tests prove that the forecast accuracy and efficiency of the optimized LSTM are higher than those of ARIMA, LightGBM, Prophet, and DeepAR. Therefore, it can provide better support for operators' resource investment. The network topology optimization model proposed in this paper is optimized based on the actual size of the network that we worked on in the study. In view of different locations, periods of time, network sizes, and network characteristics, the model needs to be optimized in accordance with different topology reshaping rules tailored based on the nodes and routing of the network to be optimized. This model is vital to network topology optimization, as it can enhance the resource utilization by over 30%.

In this paper, a brand-new AI-based network topology optimization system is proposed. By analyzing future network traffic development trends via a traffic forecast model and refining network structures through a network topology optimization model, this system enhances the utilization of network resources and reduces operators' investments. The advent of the 5G era brings the world booming network traffic and more complicated network structures, and AI-based network topology optimization is superior to other methods for its outstanding capability and practicability in dealing with complex networks. It ensures sustainable network development and guarantees the return on investment ratio for operators. The AI-based network topology optimization system introduced in this paper is proposed based on ITU AI standards, and features:

1) Creativity: It addresses difficulties and promotes the development of intelligent networks by applying cutting-edge AI technologies to operator networks.

2) Enhanced traffic forecast algorithm: It enables more accurate traffic forecasting.

3) A complete network topology analysis and optimization structure: The concepts of neighbor, node removing method, and three-step network topology optimization that are introduced for the first to the industry effectively accelerate the network topology analysis. Moreover, the network restructuring fixes the defects of existing network topologies, paving the way for future network topology optimization.

The challenge organized by ITU has offered a great opportunity for building a cross-field ecosystem, and operators and ITU should continue to make joint efforts (e.g. encourage crowd-funding AI algorithms) to resolve common problems. For example, building a middleware platform to open data and solve problems together, organizing more competitions and building the ecosystem for developers for closer collaboration.

REFERENCES

[1] Lampiris E., Zhang J., Simeone O., et al. Fundamental Limits of Wireless Caching under Uneven-Capacity Channels[J]. 2019.

[2] He Z., Jian H. Application of Multilink Aggregation and Load Balancing in Wireless Real-Time Video Transmission System[C]// 2018 International Conference on Sensor Networks and Signal Processing (SNSP). IEEE Computer Society, 2018.

[3] Wang Y., Gong B., Gong M. Research on the Trusted Energy-Saving Transmission of Data Center Network. China Communications, 2016.

[4] Kamiyama N. Network Topology Design Using Data Envelopment Analysis[C]// IEEE Globecom-ieee Global Telecommunications Conference. IEEE, 2017.

[5] Liu Z., Zou Z. Analysis of network topology and deployment mode of 5G wireless access network[J]. Computer Communications, 2020, 160.

[6] Singh J P. THE ROLE OF AI, ML, AND IOT IN DIGITAL TRANSFORMATION IN 2019[J]. PC Quest, 2019, 32(1):18-19.

[7] Liu Z., Yubo M U., Zhang Y. Overview of network artifical intelligence requirements and applications. Telecommunications Network Technology, 2018.

[8] Weytjens H., Lohmann E., Kleinsteuber M. Cash flow prediction: MLP and LSTM compared to ARIMA and Prophet[J]. Electronic Commerce Research, 2019(1).

[9] Trappey C V., Wu H Y. An evaluation of the time-varying extended logistic, simple logistic, and Gompertz models for forecasting short product lifecycles[J]. Advanced Engineering Informatics, 2008, 22(4):421-430.

[10] Greff K., Srivastava R K., J Koutník, et al. LSTM: A Search Space Odyssey[J]. IEEE Transactions on Neural Networks and Learning Systems","pubMedId":"27411231, 2017.

[11] Do We Really Need Deep Learning Models for Time Series Forecasting?, 2021.

[12] Gers F. A., Eck D., Schmidhuber J. Applying LSTM to Time Series Predictable through Time-Window Approaches[J]. Springer, Berlin, Heidelberg, 2001.

[13] Moustapha A. I., Selmic R. R. Wireless Sensor Network Modeling Using Modified Recurrent Neural Networks: Application to Fault Detection[J]. IEEE Transactions on Instrumentation & Measurement, 2008, 57(5):981-988.

[14] Sang C., MD Pierro. Improving trading technical analysis with TensorFlow Long Short-Term Memory (LSTM) Neural Network - ScienceDirect[J]. The Journal of Finance and Data Science, 2019, 5(1):1-11.

[15] Hagberg A., Schult D., Swart P. NetworkX Tutorial. 2011.

[16] Chuan W. W., Zhang B. M. A GRAPHIC DATABASE BASED NETWORK TOPOLOGY AND ITS APPLICATION[J]. Power System Technology, 2002.

[17] Chen L. W., Martin N., Cabrera C., et al. Communication network topology determination: US, US20110188409 A1[P]. 2011.

[18] Wang H., Chen Y. Network topology description and visualization[J]. IEEE, 2010.

[19] O'Brien P. S. Optimizing hand-off neighbor lists for improved system performance[J]. 2004.

[20] Sánchez-Torrubia M. G., Torres-Blanc C., Cubillo S. Design Of A Fuzzy Inference System For Automatic DFS & BFS Algorithm Learning Assessment [M]. 2015.

[21] Vijay I., Garg K. Algorithm DFS(S): begin. 2003.

[22] Yang C., Ma J., Yao Z., et al. Method for routing mobile node in wireless mesh network and a communication system thereof [J]. 2012.

[23] Wang W., Yang W., Yang C., et al. Method For Routing Mobile Node In Wireless Mesh Network And A Communication System Thereof [J]. 2008.

[24] Jinsong Gui, Kai Zhou. Flexible Adjustments Between Energy and Capacity for Topology Control in Heterogeneous Wireless Multi-hop Networks. Central South University，2016.

[25] Rui Ma, Xianghui Cao, Shuai Zhang, Lu Liu, Yu Cheng and Changyin Sun. A DBN-based Independent Set Learning Algorithm for Capacity Optimization in Wireless Networks. Southeast University，2018.

[26] Brandon Heller, Srini Seetharaman, Priya Mahadevan, Yiannis Yiakoumis, Puneet Sharma, Sujata Banerjee and Nick McKeown. ElasticTree: Saving Energy in Data Center Networks. Stanford University. 2010.

[27] Bryan Perozzi, Rami Al-Rfou, Steven Skiena. DeepWalk: Online Learning of Social Representations. Stony Brook University Department of Computer Science. 2014.

[28] Danyang Zhuo, Monia Ghobadi, Ratul Mahajan, Klaus-Tycho Förster, Arvind Krishnamurthy and Thomas Anderson. Understanding and Mitigating Packet Corruption in Data Center Networks. niversity of Washington. 2017

[29] Krzysztof Rusek, José Suárez-Varela, Albert Mestres,Pere Barlet-Ros and Albert Cabellos-Aparicio. Unveiling the potential of Graph Neural Networks for network modeling and optimization in SDN.

[30] Zhang Q., Chen G., Liang Z., et al. Topology constructing and restructuring mechanisms for Bluetooth radio networks[C]// 2016 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW). IEEE, 2016.

[31] Chiang M., Ou C., Wang J., et al. Method and apparatus for configuring a network topology with alternative communication paths: US, US7693072 B2[P]. 2010.

[32] Missaoui, Nègre, Anggraini, et al. Social network restructuring after a node removal[J]. International Journal of Web Engineering & Technology, 2013, 8(1).

AUTHORS

Han **Zengfu** received his bachelor's degree in applied physics from Beijing Jiaotong University in 2000, and his master's degree in electromagnetic field and microwave technology from the same university 5 years later. His research interests focus on machine learning theory and algorithms. In 2020, Weeny Wit, the team led by Mr. Han, won the Gold Champion (1st prize) in ITU AI/ML 5G Challenge.

Kong Jiankun received his bachelor's degree in electronic engineering from Shandong University in 1993, and his master's degree in Radio Electronics from Peking University in 1996.

Wang Zhiguo received his bachelor's degree in telecommunications engineering from Chongqing University of Posts and Communications in 2002, and his master's degree in electromagnetic fields and microwave technology from the same university in 2005. His research interests focus on mobile communications, electromagnetic field and microwave technology, and 5G network optimization, etc.

Zhang Yiwei received her bachelor's degree in electrical engineering and automation from the Northeast Forestry University in China in 2017 and her master's degree in telecommunications engineering from the University of Wollongong (UOW) in Australia in 2019. After graduation, she worked in the field of wireless communications, with her research interests focusing on network optimization. At the end of 2020, Weeny Wit, the team where Ms. Zhang was a key member, won the Gold Champion (1st prize) in ITU AI/ML 5G Challenge.

Liu Ke received his bachelor's degree in telecommunications engineering from Shandong University in 1999. His research interests include mobile communications theory and 5G network optimization, etc.

Pan Liang received his master's degree in signal and information processing engineering from Shandong University in 1999.

Li Sicong received his bachelor's degree in NanJing University of Posts and Communications of China in 2014. Since then he has been working in China Mobile Shandong Branch, responsible for LTE and 5G wireless optimization. His research interests include LTE/5G wireless networks with a focus on principles and signaling and protocols. He has been awarded by China Mobile for outstanding representation in 2017 and 2020's competition. Especially in 2020 where he won the 1st prize in the group competition, 2nd prize in the individual competition.

Wu Desheng graduated from Shandong University in 2009, and now works in the Network Department of China Mobile Shandong Branch, responsible for the maintenance and optimization of 4/5G networks.