Network Data Generation using IP2Vec Embedding

Minji Kim
Dept. of Applied Artificial Intelligence
Sungkyunkwan University
Seoul, Republic of Korea
m5512m@g.skku.edu

Migyeong Kang
Dept. of Applied Artificial Intelligence
Sungkyunkwan University
Seoul, Republic of Korea
gy77@g.skku.edu

Eunil Park
Dept. of Applied Artificial Intelligence
Sungkyunkwan University
Seoul, Republic of Korea
eunilpark@skku.edu

Sangheon Pack
School of Electrical Engineering
Korea University
Seoul, Republic of Korea
shpack@korea.ac.kr

Jinyoung Han*
Dept. of Applied Artificial Intelligence
Sungkyunkwan University
Seoul, Republic of Korea
jinyounghan@skku.edu

***Abstract*—Network data generation is crucial in applying deep learning techniques for predicting and managing 5G/6G networks. In this paper, we propose to incorporate IP2Vec in generative adversarial networks (GAN)-based network data generation models. Our evaluation demonstrates that utilizing IP2Vec can effectively enhance the performance of GANs in generating network data.**

***Index Terms*—5G/6G networks, network data generation, generative adversarial networks, IP2Vec**

I. Introduction

Deep learning techniques have been widely applied for predicting network traffic. However, deep learning models generally require a large amount of data, and especially for emerging networks such as private 5G/6G networks, it is challenging to obtain sufficient data for training [1]. Using large-scale training datasets with high variance can enhance the robustness of deep learning models for network traffic prediction tasks. Hence, generating realistic network traffic data has receive great attention, which allows to learn deep learning models for predicting network traffic [2].

Recently, generative adversarial networks (GANs) such as TimeGAN [3] and SigCWGAN [4] have demonstrated outstanding performance in generating time-series data for traffic prediction tasks. These models have shown great capabilities in capturing the temporal dependencies and characteristics of network traffic, which show great utility in generating realistic synthetic network traffic data. For example, [5] applied TimeGan for the solar power generation forecasting task, which can capture the complex temporal patterns and dependencies in solar power generation data, enabling accurate prediction for future power outputs.

The GAN-based network data generation models have been primarily designed to handle continuous input attributes [6], as traffic quantities can be represented with continuous variables [7]. However, network traffic data usually contains not only continuous information but also categorical attributes such as IP addresses, port numbers, and protocols, which are crucial in modeling network traffic, but little attention has been paid to incorporating such categorical elements in GAN-based network data generation models.

*Corresponding author.

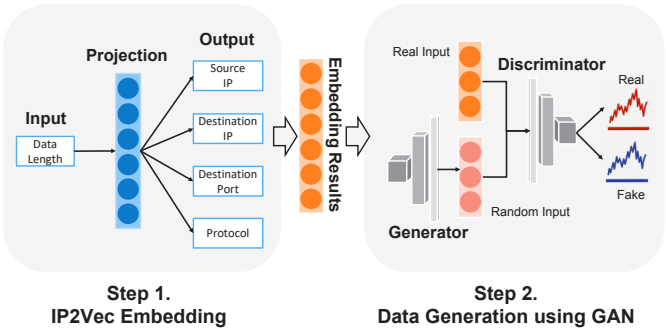

Fig. 1. The proposed model architecture for incorprating IP2Vec.

To address this problem, we propose to apply IP2Vec [8], an effective embedding technique for IP data, which can convert categorical IP attributes into continuous vectors, to GAN-based generation models. In particular, we incorporate IP2Vec into representative GAN-based models including GMMN, RCGAN, TimeGAN, and SigCWGAN, and evaluate them to investigate the effectiveness of utilizing IP2Vec in GAN-based network data generation.

II. The Model

A. Model Architecture

Figure 1 depicts the overall architecture of the model that consists of the IP2Vec embedding layer and follow-up GAN-based data generation. Initially, the IP2Vec model receives the traffic size as input and produces the corresponding embeddings as the output. Note that using IP data is challenging due to its categorical nature and lack of inherent order. Hence, to overcome these challenges, we propose to apply IP2Vec,

an unsupervised learning approach that captures the similarity between IP addresses. Here, IP2Vec can address the absence of inherent order in IP addresses by employing a skip-gram neural network architecture similar to Word2Vec. By utilizing context information from flow-based data, the model learns the similarities between IP addresses. Subsequently, the weights from the input to the hidden layer in the trained neural network can be utilized as vector representations of IP addresses.

The resulting embeddings are then used as input data for a GAN (Generative Adversarial Network), which is a generative model where a generator and a discriminator engages in an adversarial training process. The generator aims to generate synthetic data that resembles real data, whereas the discriminator tries to distinguish between real and synthetic data. Through this process, the generator gradually improves in generating data that is more similar to the real data. In this way, GANs have been actively utilized in various fields for realistic data generation such as traffic generation.

B. Time-series Generative Adversarial Network Models

In this paper, we trained and generated data using four widely-used GAN models by utilizing the dataset described in Section III-A. The descriptions of each model is summarized as follows.

- *GMMN*: GMMN is a model that effectively optimizes the Minimax loss by leveraging Maximum Mean Discrepancy (MMD). It is a variation of GANs, where MMD is used during the optimization process between the generator and discriminator to minimize the difference between distributions.
- *RCGAN*: RCGAN, which stands for Recurrent Conditional Generative Adversarial Network, is a specialized conditional generative model designed for sequence data generation. RCGAN focuses on generating time series data based on given conditions.
- *TimeGAN*: TimeGAN is a specialized model for time series data, which trains both autoencoding components and adversarial components together. It uses a Recurrent Neural Network (RNN)-based autoencoder to capture the characteristics of time series data, and generates realistic time series data through adversarial networks. TimeGAN can learn patterns and dependencies that change over time in the data.
- *SigCWGAN*: SigCWGAN is a model trained to capture the temporal dependence of conditional probabilities in time series data. It is a variation of Conditional Wasserstein GAN (CWGAN), where the optimization process between the generator and discriminator captures the temporal dependence of conditional probabilities derived from time series data.

III. Experiments

A. Dataset

In order to learn the proposed model, we employ a publicly available traffic dataset introduced by [9]. The dataset comprises network packets gathered between February 20th and October 6th in 2019. It encompasses various components such as packet timestamp, protocol, payload size, source/destination IP addresses, source/destination UDP/TCP ports, and user activity types (i.e., Interactive, Bulk, Video, and Web). The user activity types can be summarized as follows.

- *Interactive*: Interactive activity includes traffic from real-time interactive applications such as chatting app or remote file editing in Google Docs.
- *Bulk*: Bulk data transfer activity consists of traffic to applications that use significant portions of the network's bandwidth for large data file transfers, e.g., large file downloads from Dropbox.
- *Video*: Video playback activity includes traffic from applications consuming videos (e.g., from Twitch or YouTube).
- *Web*: Web browsing consists of traffic for all activities within a web page such as downloading images or ads.

TABLE I
Results of statistical analysis of datasets.

	Interactive	Bulk	Video	Web
Count	9,536	3,790	1,163	1,296
Mean	6.87	1457.21	633.60	55.16
Std	5.33	1,432.66	946.86	174.98
Min	1.00	1.00	1.00	1.00
25%	3.00	348.00	2.00	2.00
50%	6.00	955.50	13.00	5.00
75%	9.00	2365.00	1250.00	18.00
Max	48.00	8129.00	7562.00	2037.00

Statistical analysis was conducted to figure out the characteristics of each type of dataset, and the results are presented in Table I. In the Interactive and Web types, the standard deviations were relatively low, with values of 5.33 and 174.98, respectively, indicating minimal differences among the data. However, for the Bulk and Video types, the standard deviations were approximately 5 to 60 times higher, revealing significant variations between the datasets. In our experiments, we used these four datasets based on user activity types and applied the GAN models described in Section II-B to generate additional data. We then evaluated the performance of the synthetic data.

B. Evaluation Metrics

To evaluate the traffic generation performance of the proposed model, we employ two widely-used metrics: (i) R^2 Comparison (R-S), (ii) Discriminative Score (DS).

- *R^2 Comparison*: Assuming that we would like to predict the next-step temporal vectors using the lagged values of time series. To evaluate the quality of the synthetic data, we first train a supervised learning model on real data and assess its performance on real data by measuring the R^2 (TRTR). Subsequently, we train the same supervised learning model on synthetic data and calculate the R^2 of the trained model on the real data (TSTR). The proximity of these two R^2 indicates the quality of the generative model, with a smaller difference suggesting a better performance.

- *Discriminative Score*: To quantify the similarity, we employ a post-hoc time-series classification model, trained by optimizing a two-layer LSTM (Long Short-Term Memory), to differentiate between sequences derived from the original dataset and those generated by the model. Initially, each sequence from the original dataset is labeled as 'real' while each generated sequence is labeled as 'not real'. Subsequently, an RNN (Recurrent Neural Network) classifier is trained to distinguish between these two classes in a standard supervised task. Finally, we report the classification accuracy on the separate test set to assess the performance of the model.

C. Experimental Results

Table II presents the performance of the GMMN, RCGAN, TimeGAN, and SigCWGAN models combined with IP2Vec for the four types of data: interactive, bulk, video, and web. In addition, for comparison purposes, the performance of models without using IP2Vec is also measured. As shown in Table II, both R^2 comparison and discriminative score generally exhibited lower values when using IP2Vec compared to not using it in most cases. Particularly, the GMMN and SigCWGAN models demonstrated superior performance in terms of discriminative score when IP2Vec was applied for all data types. When examining the performance for each data type, web data showed the largest improvement in terms of R^2 comparison compared to other data types. Specifically, when the GMMN model was trained on web data, it achieved a value of 0.76, but when IP2Vec was introduced, it showed a performance of 0.22, resulting in a remarkable performance difference of 0.54. This result indicates that even models like GMMN, which do not capture the characteristics of time series data well, can achieve more effective data generation by leveraging IP2Vec.

TABLE II
PERFORMANCE RESULTS OF EACH MODEL AND DATA WITH AND WITHOUT IP2VEC.

Model	Embedding	Type	TSTR	R-S	DS
GMMN	w/o IP2Vec	Inter	0.643	0.265	0.367
		Bulk	0.908	0.063	0.500
		Video	0.294	0.622	0.185
		Web	0.149	0.760	0.335
	w IP2Vec	Inter	0.765	**0.135**	**0.083**
		Bulk	0.803	0.157	**0.360**
		Video	0.759	**0.137**	**0.169**
		Web	0.703	**0.218**	**0.070**
RCGAN	w/o IP2Vec	Inter	0.667	0.241	0.388
		Bulk	0.903	0.068	0.483
		Video	0.674	0.242	0.096
		Web	0.588	0.321	0.380
	w IP2Vec	Inter	0.681	**0.220**	**0.145**
		Bulk	0.855	0.105	**0.278**
		Video	0.644	0.252	0.208
		Web	0.668	**0.253**	**0.119**
TimeGAN	w/o IP2Vec	Inter	0.720	0.188	0.355
		Bulk	0.909	0.062	0.494
		Video	0.682	0.234	0.073
		Web	0.651	0.258	0.339
	w IP2Vec	Inter	0.721	**0.180**	**0.145**
		Bulk	0.868	0.091	**0.219**
		Video	0.680	**0.216**	0.140
		Web	0.716	**0.205**	**0.121**
SigCWGAN	w/o IP2Vec	Inter	0.883	0.025	0.417
		Bulk	0.947	0.024	0.492
		Video	0.895	0.021	0.112
		Web	0.856	0.053	0.307
	w IP2Vec	Inter	0.886	**0.014**	**0.116**
		Bulk	0.942	**0.018**	**0.216**
		Video	0.870	0.027	**0.107**
		Web	0.916	**0.005**	**0.128**

IV. CONCLUSION

In this paper, we investigated the performance of various GANs for generating time-series traffic data. To this end, we proposed to apply IP2Vec, which can effectively embed IP information into the model. For evaluation, we employ an open source dataset comprising network packets collected from February 20th to October 6th in 2019. The results demonstrated that GANs with IP2Vec show higher performance than the baselines in situations with limited data availability. This research has the potential to assist in network resource management in private 5G/6G networks, particularly for emerging services where sufficient data may not be available.

ACKNOWLEDGMENT

This research was supported by the National Research Foundation (NRF) of Korea Grant funded by the Korean Government (MSIT) (No. 2021R1A4A3022102) and by the MSIT (Ministry of Science and ICT), Korea, under the ICAN (ICT Challenge and Advanced Network of HRD) support program (IITP-2023-RS-2023-00259497) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation).

REFERENCES

[1] Sicker Douglas C., Ohm Paul, and Grunwald Dirk. 2007. Legal issues surrounding monitoring during network research. In Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement. ACM, New York, NY, 141–148. DOI: https://doi.org/10.1145/1298306.1298307

[2] Adeleke, Oluwamayowa Ade, Nicholas Bastin, and Deniz Gurkan. "Network traffic generation: A survey and methodology." ACM Computing Surveys (CSUR) 55.2 (2022): 1-23.

[3] Jinsung Yoon, et al. "Time-series generative adversarial networks". In Advances in Neural Information Processing Systems, pages 5509–5519, 2019.

[4] Ni, Hao, et al. "Conditional sig-wasserstein gans for time series generation." arXiv preprint arXiv:2006.05421 (2020).

[5] Li, Qing, et al. "A Multi-step ahead photovoltaic power forecasting model based on TimeGAN, Soft DTW-based K-medoids clustering, and a CNN-GRU hybrid neural network." Energy Reports 8 (2022): 10346-10362.

[6] Ring, Markus, et al. "Flow-based network traffic generation using generative adversarial networks." Computers & Security 82 (2019): 156-172.

[7] Kang, Chaewon, et al. "TransTraffic: Predicting Network Traffic using Low Resource Data." 2022 13th International Conference on Information and Communication Technology Convergence (ICTC). IEEE, 2022.

[8] Ring, Markus, et al. "Ip2vec: Learning similarities between ip addresses." 2017 IEEE International Conference on Data Mining Workshops (ICDMW). IEEE, 2017.

[9] V. Labayen, E. Magana, D. Morat ˜ o, and M. Izal, "Online classification of user activities using machine learning on network traffic," Computer Networks, vol. 181, p. 107557, 2020.