Generative Adversarial Learning for Machine Learning empowered Self Organizing 5G Networks

Ben Hughes*, Shruti Bothe†, Hasan Farooq†, Ali Imran†
*Booker T. Washington High School, Tulsa, USA
†University of Oklahoma, Tulsa, USA
Email: hughesbenm@gmail.com*,{shruti, hasan.farooq, ali.imran}@ou.edu†

***Abstract*—In the wake of diversity of service requirements and increasing push for extreme efficiency, adaptability propelled by machine learning (ML) a.k.a self organizing networks (SON) is emerging as an inevitable design feature for future mobile 5G networks. The implementation of SON with ML as a foundation requires significant amounts of real labeled sample data for the networks to train on, with high correlation between the amount of sample data and the effectiveness of the SON algorithm. As generally real labeled data is scarce therefore it can become bottleneck for ML empowered SON for unleashing their true potential. In this work, we propose a method of expanding these sample data sets using Generative Adversarial Networks (GANs), which are based on two interconnected deep artificial neural networks. This method is an alternative to taking more data to expand the sample set, preferred in cases where taking more data is not simple, feasible, or efficient. We demonstrate how the method can generate large amounts of realistic synthetic data, utilizing the GAN's ability of generation and discrimination, able to be easily added to the sample set. This method is, as an example, implemented with Call Data Records (CDRs) containing the start hour of a call and the duration of the call, in minutes taken from a real mobile operator. Results show that the method can be used with a relatively small sample set and little information about the statistics of the true CDRs and still make accurate synthetic ones.**

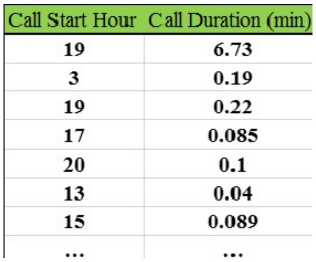
***Index Terms*—Deep learning, Generative Adversarial Network, Synthetic data, CDRs, 5G, SON.**

I. Introduction

While the final image of the future mobile cellular radio access network is yet to emerge, network densification, miscellany of nodes, split of control and data plane, network virtualization, heavy and localized cache, infrastructure sharing, concurrent operation at multiple frequency bands, simultaneous use of different medium access control and physical layers, and flexible spectrum allocations can be envisioned to be some of the potential ingredients of future MCN [1]. It is not difficult to prognosticate that with such conglomeration of technologies, the complexity of operation and resultant resource inefficiency and shrinking profit margins are to become the biggest challenges for MCN operators. This means automation of the post-deployment operation and optimization in MCN for reducing costs, handling complexity and maximizing resources efficiency will not only become a necessity, but the future MCN's technical and commercial viability may hinge on it.

As the root of adaptability stems from intelligence, it is no wonder that machine learning is perceived as the panacea for many of the challenges being faced by future cellular networks. In the context of mobile networks, advanced machine learning methods can leverage big data to model spatiotemporal network behavior that in turn can be used to self-configure, self-optimize, self-heal and self-manage the network with no or minimal human involvement, dubbed as Self-Organizing Network (SON) [2]. Machine learning algorithms, driving the SON functions, can predict future network state for pre-allocating network resources more intelligently and in a more efficient manner and thus transforming all reactive style SON functions into proactive ones for meeting ambitious quality of service requirements for 5G. Machine learning algorithms require large amounts of true training data as presence of more data results in better and accurate models since it allows the data to tell for itself instead of relying on assumptions and weak correlations. A weak assumption coupled with complex algorithms is far less efficient than using more data with simpler algorithms. This fact has been captured by many studies, e.g., [3], [4] wherein results suggests for a given problem, adding more examples to the training set monotonically increases the accuracy of the model. Due to this, SON fueled by machine learning algorithms needs an incredible amount of true data, often in very different data types. However, one of the key challenges faced by this approach is data scarcity issue since labelled real data is often not readily available.

To fully address the data scarcity/sparsity problem at a more fundamental level, we propose a solution to this using data-driven (or model-free) approach by adopting generative methods. Specifically, we propose to utilize the power of the recently discovered machine learning concept of Generative Adversarial Networks (GANs) [5]. The key benefit of GAN is that they can directly generate new datasets based on historical data, without explicitly specifying a model or fitting probability distributions. GANs have been proven to be effective at generating realistic data in the format of images, graphs, and speech [6]–[8]. In the image processing community, GANs are able to generate realistic images that are of far better quality compared to other methods. Few works exist where GANs found their applications in other domains like in [9]



Call Start Hour	Call Duration (min)
19	6.73
3	0.19
19	0.22
17	0.085
20	0.1
13	0.04
15	0.089
...	...

Fig. 1. Real CDRs Training Dataset for GAN

for scenario generation used in power systems and in [10] that implemented GAN for spectrum sensing and bench marked its effectiveness. The results showed GAN based approach is a promising candidate for generating realistic synthetic datasets. Inspired by the usefulness of GANs reported by these studies, in this paper, we have also leveraged GAN for generating synthetic calling patterns (CDRs) of mobile users. It is important to mention here that aforementioned relevant studies generated datasets other than the mobile network traffic traces. Therefore, conclusions drawn from these studies cannot be directly applied to mobile network traces such as CDRs. Study presented in this paper fills this gap.

The main contributions of this paper can be summarized as follows:

1) In this work we propose that in realm of cellular networks, GAN can be leveraged to generate synthetic tabular data, in the form of Call Data Records (CDRs) and increase dataset size by augmenting the real dataset with realistic synthetic data. This GAN based approach is significantly different from conventional approaches of first fitting a model using historical observations, and then the fitted probabilistic models are sampled to generate new synthetic datasets. The dynamic and time-varying nature of CDRs coupled with the complex spatial and temporal interactions make model-based approaches difficult to apply and hard to scale. A single set of model parameters normally cannot capture complex mobile network dynamics, especially when multiple users are considered. These models are typically constructed based on statistical assumptions that may not hold or difficult to test in practice. To the best of authors' knowledge, this is the first time that GAN algorithm has been used to generate synthetic CDRs. In this work we chose CDRs generation as an example case study since they are required by majority of the SON algorithms such as in [11], [12].
2) We have leveraged real network traces (snapshot of dataset shown in Fig. 1) with start hours of calls and the calls durations provided by one of leading mobile operators in USA as the two dataset features to train the GAN. Therefore, the performance of GAN tested using real traffic traces truly represent its performance with real network data.
3) We also augmented GAN generated synthetic CDRs with real ones to increase size of training dataset and evaluated how well the deep neural network predicts call duration times and bench marked against dataset having real CDRs only. Results showed that by augmenting real data set with GAN generated synthetic data set, we can expect improved prediction accuracy of ML algorithms driving proactive SON functions for future 5G networks.

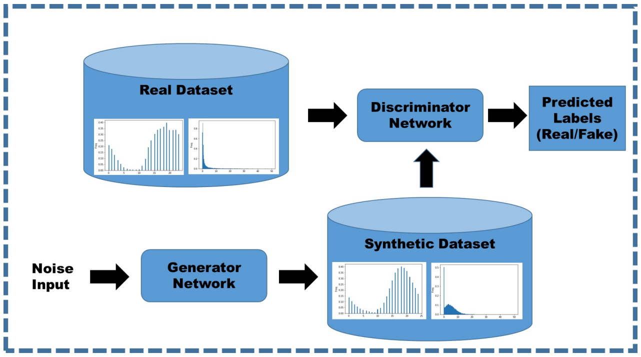


Fig. 2. GAN Structure

The rest of the paper is organized as follows: Section II describes the GAN model; Results are illustrated and evaluated in Section III; and Section IV concludes the paper.

II. The Adversarial Model

GANs are a recent development in the field of machine learning, the initial research paper that introduced them was published in 2014 [5]. Despite this, they have already been used for a multitude of subjects, such as increasing the resolution of a pixilated photo and predicting the next words in a sentence. The intuition behind GANs is to leverage the power of deep neural networks to both express complex nonlinear relationships (the generator) as well as classify complex signals (the discriminator). All GANs more or less follow the same structure, two artificial neural networks (generator DNN and the discriminator DNN) working together in a minimax two player game (Fig. 2) (thus the use of adversarial in the name).

Let $\mathbb{P}_X$ be the true distribution of the observation, which is unknown and hard to model. Of the two networks in our GAN, one is the generator, designed to take some random noise with distribution $\mathbb{P}_Z$ as an input and produce fake data $G(Z)$ with a distribution $\mathbb{P}_G$ that it then passes to the second network. This second network is the discriminator, designed to be pre-trained on the true sample data with distribution $\mathbb{P}_X$ and produce some value of how real data is expressed as $D(x)$ where x may come from $\mathbb{P}_X$ or $\mathbb{P}_Z$. The discriminator is trained to distinguish between $\mathbb{P}_G$ and $\mathbb{P}_X$, and thus to maximize the difference between $\mathbb{E}[D(X)]$ (real data) and $\mathbb{E}[D(G(Z))]$ (generated data). A large discriminator output means the sample is more realistic, therefore the generator tries to maximize the value of $\mathbb{E}[D(G(Z))]$. The loss function L_G for the generator therefore becomes $L_G = -\mathbb{E}_Z[D(G(Z))]$. In the same vein, the discriminator tries to minimize the value of $\mathbb{E}[D(G(Z))]$, the loss function L_D for the discriminator then becomes $L_D = -\mathbb{E}_X[D(X)]+ \mathrm{E}_Z[D(G(Z))]$. Finally a two-player minimax game is set up between G and D with the value function $V(G, D)$:

$$\min_G \max_D V(G, D) = \mathbb{E}_X[D(X)] - \mathbb{E}_Z[D(G(Z))] \quad (1)$$

where $V(G, D)$ is the negative of L_D. In the initial stages of training, G just generates CDR data samples $G(z)$ totally different from samples in $\mathbb{P}_X$, and discriminator can reject these samples with high confidence. Gradually the generator learns to generate samples that could let D output high confidence to be true, while at the same time the discriminator is also trained to distinguish these newly fed generated samples from G. As training moves on and goes near to the optimal solution, G is able to generate samples that look as realistic as real data with a small L_G value, while D is unable to distinguish $G(z)$ from $\mathbb{P}_X$ with large L_D [9]. In theory, at reaching the Nash equilibrium, the optimal solution of GANs provide such a generator that can exactly recover the distribution of the real data so that the discriminator would be unable to tell whether a sample came from the generator or from the historical training data. At this point, generated traces are indistinguishable from real historical data, and are thus as realistic as possible.

While all GANs have this same format, they all likely have slightly different configurations of activation functions, hidden layers, and neuron counts. Most include deep artificial neural networks as the discriminator and generator and one hundred input neurons for random noise. Activation functions are another variable in the structure of a GAN. It is almost essential for the generator to be activated with a rectified linear unit (ReLU) function, but the discriminator can range from ReLU, to a hyperbolic tangent, to a sigmoid function, or any combination of the three or more. The GAN we used to test our proposed method was fairly standard, with both integrated networks being deep artificial neural networks and using both ReLU and sigmoid functions as activations. Unlike most GANs, which often use massive arrays for pictures and graphs, ours only had start hour and call duration as its inputs derived from CDRs provided by one of the national mobile operators in USA. To counter that, we increased the random noise from one hundred random values to two hundred, finding that fewer often forced the networks into a rut with zero progress.

III. Results & Discussion

Starting with twenty thousand data points (from a list of several hundred thousand), we trained the discriminator. After the discriminator could reliably tell the difference between true data taken from the list and randomly generated data we started training the generator. Once the generator was making data that the discriminator thought was real we generated another twenty thousand data points. Figs. 3 and 4 represent the frequency distribution of the sample set we used to train the discriminator.

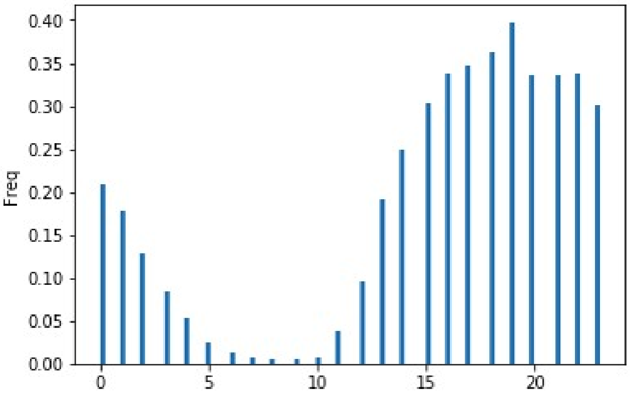


Fig. 3. Real CDRs Calls Start Hour Histogram

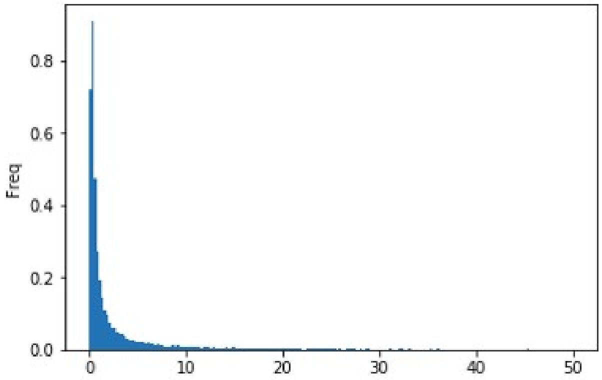


Fig. 4. Real CDRs Calls Duration Histogram with x-axis in minutes

As per the figures, it is evident that most calls originate in afternoon-evening time with majority of the calls having short duration in order of few minutes. Figs. 5 and 6 represent the frequency distribution of the twenty thousand synthetic data points generated by the generator after the GAN was fully trained with real CDRs. As the figures show, our method was more than capable of making realistic synthetic data. The generated data is not perfect; there are some inconsistencies,

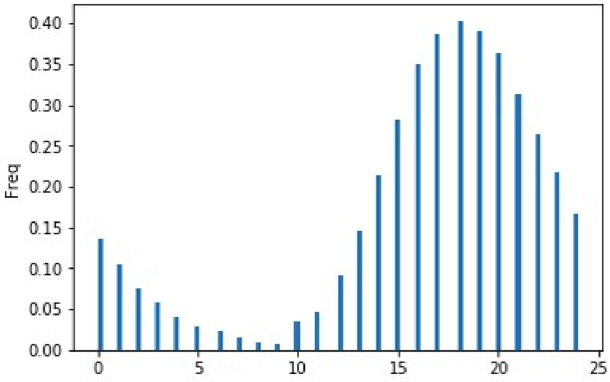


Fig. 5. Synthetic CDRs Calls Start Hour Histogram

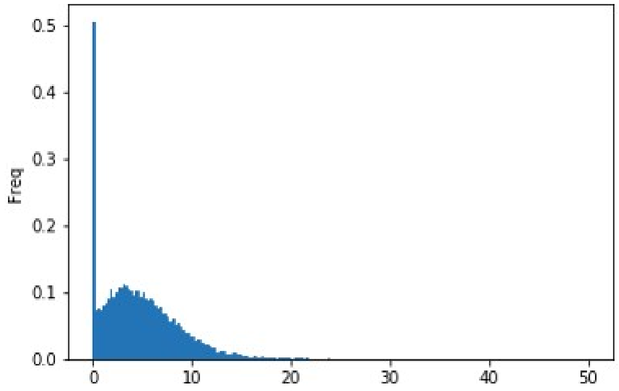


Fig. 6. Synthetic CDRs Calls Duration Histogram with x-axis in minutes

but for the most part the synthetic data could easily be added onto the sample set for augmentation. The biggest success of this method is the distribution of synthetic start hours. The specific shape of the graph that Fig 5 shows is the result of another method within our work. By shifting the "start" and "finish" of the day by eight hours, Fig. 3 showed a very pleasant bell curve. By performing that shift before training, the generator learned to make a similar curve. After the generation of the synthetic data was finished, a simple eight-hour reverse shift produced the shape in Fig. 5. This method is only possible and accurate due to the nature of a day, that while according to our measurement of time it has a start and finish, the hours in a day are continuous and can be manipulated. Another detail was the ranges of the generated datasets. Start hours of the calls must naturally be within 0 and 23, and while Fig. 5 and its data include many of the value 24, the fact that there are zero other outliers is a success. Call duration had the opposite results. Fig. 4 shows outliers upwards of 30 and 40 minutes, while Fig. 6 shows none.

The Real CDRs start hours exhibited mean value of 15.4 with variance of 45.2 while synthetic CDRs start hours showed mean value of 15.1 with variance of 43.2. The real CDRs had a mean value of 3.4 minutes for call duration with variance of 97.1 while synthetic CDRs showed mean value of 4.8 minutes for call duration with variance of 14.8. This shows that synthetic mean is very close to real mean. The variances of start hour are likewise similar, but the variance of synthetic duration is much lower than that of real duration, clearly due to the lack of large outliers in the set. The reason for this error seems to be the normalization of the data sets previously mentioned. The artificial neural networks need reliable ranges and massive outliers go against this. This also means that the generator will likely never produce values like the outliers, simply because the network has learned that they are fake data, even when they exist in the training set.

Next, we augmented GAN generated synthetic CDRs with real ones to increase size of training dataset and evaluated how well the deep neural network (DNN) predicts call duration times and bench marked against dataset having real CDRs only. Real data set was divided into training and testing sets in the ratio of 70:30 with only the training set being used for GAN. The neural networks had 8 hidden layers and used 'ReLU' as the activation function. The results are shown in Fig. 7 according to which with real data only, DNN achieved prediction accuracy of about 84.23%. Once we augmented the real CDRs with synthetic data, DNN achieved higher accuracy of about 87.86%. These results make promising revelation about the fact that by augmenting real data set with GAN generated synthetic data set, we can expect improved prediction performance of ML algorithms driving proactive SON functions for future 5G networks [12].

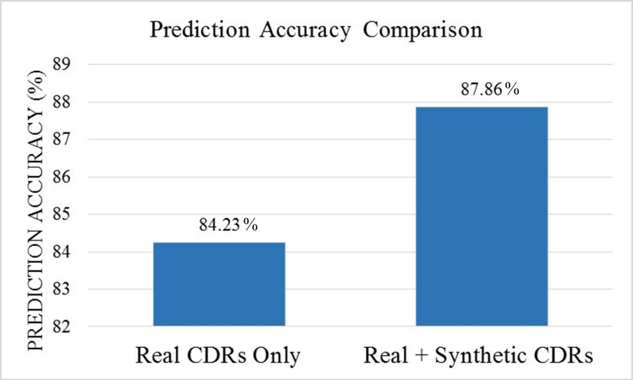


Fig. 7. Call Duration Prediction Accuracy Comparison

IV. Conclusions & Future Works

The fact that SONs need a large amount of sample data to train their machine learning algorithms is undisputed. To combat that, we propose a new method of data augmentation using GANs to generate realistic synthetic data. In this paper we have described the method we used to test such a proposal on two features of CDRs and shared our successful results. From these results it seems incredibly efficient to use GANs in this way and it is sure that once the generator is trained an infinite amount of synthetic data can be generated. Depending on exactly how much of an effect training sample size has on accuracy, this method could be not only an incredibly efficient way to augment data, and therefore accuracy, but one of the best. Using what we have learned from a simple two feature dataset, it should be both possible and simple to add more features to what already has been proven possible.

Adversarial learning is still a new method of machine learning and its possibilities are similarly unexplored. This method has proven to be a novel and effective way of tackling the problem that SON development faces. With our method we effectively doubled the amount of training sample data we started with.

For future works, we will gauge performance of proactive SON algorithms being driven by GAN generated synthetic CDRs. In addition to CDRs, we will leverage GANs to generate other kind of network data like fault alarms that are helpful for self-healing functions as well as mobility pattern of the users for predictive mobility handover algorithms.

Acknowledgement

This work was made possible by NSF Grant No. 1619346. The statements made herein are solely the responsibility of the authors. For details, visit www.ai4networks.com.

References

[1] A. Imran and A. Zoha, "Challenges in 5G: how to empower SON with big data for enabling 5G," *IEEE Network*, vol. 28, no. 6, pp. 27–33, nov 2014.

[2] O. G. Aliu, A. Imran, M. A. Imran, and B. Evans, "A Survey of Self Organisation in Future Cellular Networks," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 1, pp. 336–361, 2013.

[3] M. Banko and E. Brill, "Scaling to Very Very Large Corpora for Natural Language Disambiguation," in *Proc. of ACL*, 2001.

[4] R. B. David and M. Last, "Context-Aware Location Prediction," in *Revised Selected Papers From the 5th International Workshop on Big Data Analytics in the Social and Ubiquitous Context*, vol. 9546, 2016, pp. 165–185.

[5] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative Adversarial Nets," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2014, pp. 2672–2680.

[6] A. Radford, L. Metz, and S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks," *CoRR*, vol. abs/1511.06434, 2015.

[7] S. Pascual, A. Bonafonte, and J. Serrà, "Segan: Speech enhancement generative adversarial network," in *INTERSPEECH*, 2017.

[8] Skymind. A Beginner's Guide to Generative Adversarial Networks (GANs). [Online]. Available: https://skymind.ai/wiki/generative-adversarial-network-ganresources-for-generative-networks

[9] Y. Chen, Y. Wang, D. Kirschen, and B. Zhang, "Model-free renewable scenario generation using generative adversarial networks," *IEEE Transactions on Power Systems*, vol. 33, no. 3, pp. 3265–3275, May 2018.

[10] Y. E. S. Kemal Davaslioglu, "Generative Adversarial Learning for Spectrum Sensing," in *Proc. IEEE ICC'18*, 2018 (accepted for publication).

[11] A. Zoha, A. Saeed, H. Farooq, A. Rizwan, A. Imran, and M. A. Imran, "Leveraging intelligence from network cdr data for interference aware energy consumption minimization," *IEEE Transactions on Mobile Computing*, vol. 17, no. 7, pp. 1569–1582, July 2018.

[12] H. Farooq and A. Imran, "Spatiotemporal Mobility Prediction in Proactive Self-Organizing Cellular Networks," *IEEE Communications Letters*, vol. 21, no. 2, pp. 370–373, feb 2017.