Effects of Adversarial Attacks on Time-Series Forecasting Tasks

Göktuğ Öcal
Department of Computer Engineering
Bogazici University
Istanbul, Turkey, 34342
goktug.ocal@boun.edu.tr

***Abstract*—Since the decision-making process in some businesses highly relies on Deep Neural Networks and there are lots of intelligent systems that use DNNs, the robustness of the DNNs is becoming more and more important every day in nearly every domain. Recent studies have shown that DNNs are vulnerable to perturbated attacks called adversarial attacks, which makes them non-robust. In the research community, the main example of the importance of the robustness of DNNs is the autonomous cars and car crashes. Yet there are other applications of DNNs such as time series forecasting which is very crucial for decision-making in businesses. In this study, we have investigated the effects of adversarial attacks on different DNN models, mostly based on Long Short-Term Memory(LSTM) networks.**

I. Introduction

Deep Neural Networks (DNN) and other Machine Learning methods are now essential tools for a variety of tasks like speech recognition, image classification, and natural language processing. A DNN is simply a collection of layers of neurons which are the components that take input from previous units and executes a basic computation with its own activation function. The network's neurons work together to accomplish a complicated nonlinear mapping from input to output [1]. Back-propagation is a method for changing each neuron's weight by observing errors after learning this mapping from the data. Neuron and layer units can be used for establishing different structures in order to perform different tasks. In several cases, DNNs have reached exceptionally high predicted accuracy when compared with human ability.

Another application of DNNs is time-series tasks. Modern lifestyles would certainly become more comfortable if the time-series data could be accurately predicted [2]. Time-series data are slightly different than other types because of the sequential relationship between time steps. Therefore time-series modeling requires specialized techniques to interpret the data or learn from data. DNNs are learning patterns, relations, and knowledge from data and the deep learning approach has become one of the most robust techniques for time-series modeling.

Applications of time-series modeling or prediction appear in various domains since the insight that can be extracted from the time-series data, especially insights about the future, is very valuable and requested. Stock market prediction [3] is a highly requested task, which is both rely on time-series forecasting and pattern classification, in the finance domain; energy demand forecasting [4] is another application that energy suppliers try to estimate in order to build strategies; predictive maintenance and failure prediction is a time-series classification task which is very crucial in manufacturing. All of those applications of time-series modeling simply end with decision-making made by human or autonomous systems, and the vulnerability of the models can cause crucial results. Therefore performance, not only in terms of accuracy but also robustness, of built models should be investigated and tested with techniques in the literature. Adversarial attacks which is generating perturbed samples that cause prediction failure can help to identify weaknesses in the models and provide insights into how they make their predictions. This can be useful for improving the performance of the models and making them more robust to different types of inputs. Additionally, understanding the vulnerabilities of these models can help to prevent their misuse in applications where accurate predictions are critical, such as in healthcare or finance. In the literature, adversarial attack generation is studied and applied to various models, especially in the autonomous driving domain.

In this study, we have focused on the time-series forecasting tasks and the effect of adversarial attacks

on the task. The two most used adversarial sample generation methods called Fast Gradient Sign Method (FGSM) and Projected Gradient Descent (PGD) is selected to generate perturbed samples to test forecasting models. To accomplish the forecasting task, 4 different datasets in distinct domains and multiple state-of-art DNN models are used, which are based on Long Short-Term Memory(LSTM) networks and Convolutional Neural Networks(CNN). The experiments are conducted to observe the forecasting performance of the models for different adversarial attack settings.

II. Related Work

Adversarial attacks on neural networks are attempts to deliberately cause the network to make mistakes or fail by feeding it input that is specifically designed to be difficult for the network to handle. For their generation, a number of algorithms have been created, and a number of defenses have been put out [5]. Szegedy et al. [6] were the first ones that presented the vulnerability of neural networks against adversarial attacks in 2014. The first examples was on the image classification domain [7] and most of the researches were focused on autonomous driving domain [8], [9]. To generate adversarial attacks different techniques have been proposed. In 2014, Goodfellow et al. [10] proposed an adversarial sample generation algorithm called Fast Gradient Sign Method(FGSM) which is an alternative to expensive optimization techniques and based on fast gradient. Madry et al. [11] introduced a better adversarial attack method called Projected Gradient Descent(PGD) attack which is an enhanced version of FGSM.

In the time-series forecasting domains there are numerous studies from traditional statistics based approaches [12]–[14] to deep learning approaches [15], [16] and also statistics and neural network combinations [17] have been studied as well. Recently, DNN based time-series forecasting models which are so popular are being used widely. Chimmula et al. [18] tried to predict Covid-19 transmission and ending point of the pandemic using LSTM networks which are proposed by Hochreiter et al. [19] in 1997. Jaseena et al. [20] used Bidirectional LSTM(BDLSTM) model to predict wind speed. In 2020, Livieris et al. [21] used both CNN and LSTM networks and combined them to predict gold prices.

In terms of attack on the time-series DNN models with adversarial samples, there are a couple of researches as well. Karim et al. [22] proposed a proxy attack strategy on a target time-series classifier as a black box attacking technique. Fawaz et al. [23] utilized FGSM and Basic Iterative Method(BIM) in order to generate adversarial examples that can fail the time-series classification model. Controversially, Abdu-Aguye et al. [24] studied detecting adversarial examples generated by FGSM and BIM. These ones are the attacks on classification models but there are a couple of studies that worked on forecasting tasks. In 2022, Wu et al. [25] proposed and formulated a time-series prediction adversarial attack problem and also proposed a perturbation algorithm which they have used on LSTNet model. Mode et al. [26] applied adversarial attacks, FGSM and BIM, to LSTM and CNN networks and worked on datasets from two different domains.

In our work, we wanted to utilize FGSM and PGD attacks on LSTM, BDLSTM, and CNN networks in order to observe the forecasting accuracy and robustness of the models.

III. Background

A. LSTM Models

1) Vanilla LSTM: Long short-term memory (LSTM) is a type of recurrent neural network (RNN) that is capable of learning long-term dependencies in data. LSTM is able to remember information for long periods of time, while still being able to learn and adapt to new inputs. This is achieved through the use of gating mechanisms within the network, which allow it to selectively retain or forget information as needed. With the gates and memory cell, LSTM can learn long-term dependencies. An LSTM consists of a memory cell, input gate, output gate, and a forget gate. LSTMs are commonly used in a variety of applications, including natural language processing, speech recognition, and time series forecasting.

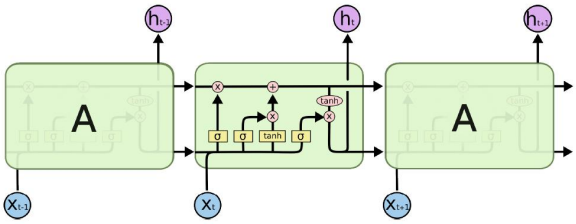


Fig. 1: An LSTM Cell

2) BDLSTM: Bidirectional long short-term memory (LSTM) is a variant of the LSTM network that processes the input sequence in both forward and backward directions. This allows the network to consider the context of a given input not just in terms of the preceding input elements, but also in terms of the subsequent input elements. This can be useful in situations where the

order of the input sequence is important, but the exact position of an input element within the sequence is less so. For example, a bidirectional LSTM might be used in a natural language processing task to accurately predict the part of speech of a given word in a sentence, taking into account both the words that come before and after it.

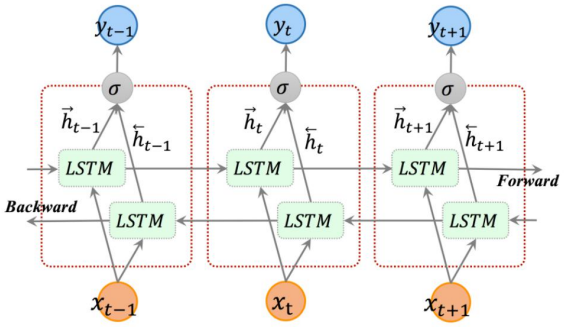


Fig. 2: A BDLSTM Structure [27]

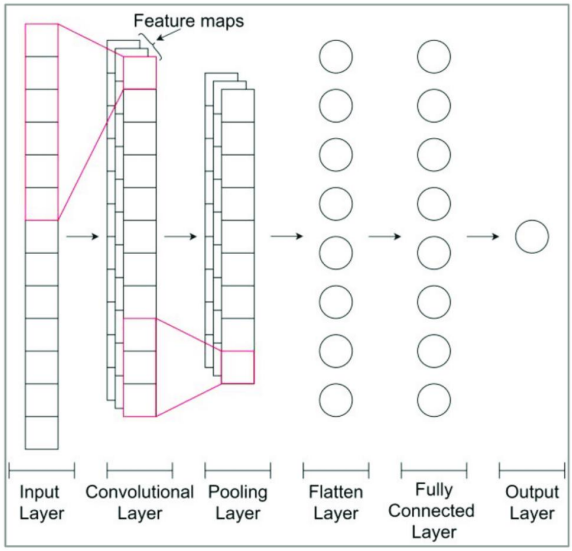


Fig. 3: CNN usage for forecasting [28]

B. CNN Models

A CNN is a type of artificial neural network that is commonly used in image and video recognition tasks. It uses a mathematical operation called convolution to filter the input data and extract features from it. This operation is performed on the input data using a set of learnable filters, called kernels or filters, which are applied at every spatial location of the input. The filters are adjusted during the training process in order to extract the most useful features from the input data. The extracted features are then passed through one or more fully connected layers, which perform the final classification or regression task. Because CNNs are able to automatically learn the most useful features from the input data, they are able to achieve high accuracy on a time series forecasting task.

CNN is popular in image classification tasks and not very common in the time-series domain but with a 1-dimensional approach as shown in Figure 3.

C. Adversarial Attacks

1) FGSM: The FGSM is a technique for generating adversarial examples, which are inputs to a machine learning model that have been specifically designed to cause the model to make mistakes. The FGSM works by adding a small, carefully calculated perturbation to the input data in the direction of the gradient of the loss function with respect to the input. The gradient tells us how the loss function changes as we vary the input, so by adding a perturbation in the direction of the gradient, we can cause the model to make a larger error on the input than it would otherwise. The FGSM is called "fast" because it is relatively simple and efficient to implement, making it a popular choice for generating adversarial examples.

$$\hat{X} = X + \epsilon * sign(\nabla_x J(\theta, x, y)) \tag{1}$$

2) PGD: Projected gradient descent (PGD) is an iterative method for generating adversarial examples, which are inputs to a machine learning model that have been specifically designed to cause the model to make mistakes. The PGD works by taking small steps in the direction of the gradient of the loss function with respect to the input and then projecting the resulting perturbation back onto the space of valid inputs. This projection step ensures that the adversarial example remains within the range of input values that the model was trained on, and prevents the perturbation from becoming too large and detectable. By iterating this process multiple times, the PGD can generate highly effective adversarial examples that are difficult for the model to detect.

$$X^t = X^{t-1} + \alpha * sign(\nabla_x L(\theta, X^{t-1}, y)) \tag{2}$$

where,

$$\alpha * sign = \begin{Bmatrix} \epsilon & if \alpha * sign > \epsilon \\ -\epsilon & if \alpha * sign < -\epsilon \\ \alpha * sign & else \end{Bmatrix} \quad (3)$$

IV. Experiments

In the experiments, four different datasets have been gathered and for each dataset 4 different forecasting models have been utilized. After all the models had been trained, FGSM and PGD attacks applied to the samples with different settings (in terms of epsilon and alpha values) and different forecasting metrics have been used in order to compare the performance of each model.

A. Many-to-one and many-to-many

Regardless from experiment setup and utilized models and attacks, two types of forecasting approaches have been conducted by their output size; many-to-one and many-to-many. In many-to-one approach regression task is conducted using a window in time series data, denoted as $[t_{-N}....t_{-1}]$ where N is the window length, as an input to the model and the model gives one output that corresponds to the next value after the window denoted as t_0. The many-to-one model is visualized in Figure 4.

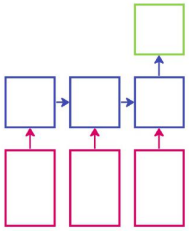


Fig. 4: Many-to-one

In many-to-one approach, this time regression task is conducted with the same window as an input to the model and the model gives a series of output denoted as $[t_0, ..., t_M]$ where M is the output range, that corresponds to the next range of values come after the window. The many-to-one model is visualized in Figure 5.

B. Models

As mentioned earlier multiple models have been used in experiments. The used models can be summarized as;

- Single Layer LSTM Model
- Double Layer LSTM Model
- Bi-directional LSTM (BDLSTM) Model
- 1D Convolutional Neural Network (1D-CNN) Model

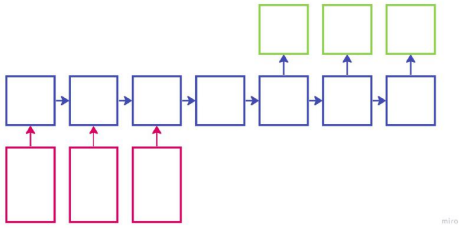


Fig. 5: Many-to-many

Single Layer LSTM Model has a single layer of LSTM cells and a dropout layer after that layer and a dense layer produces the final output. The LSTM layer has tanh activation function and 32 neurons in the hidden state, and the final dense layer has the number of neurons by output size. The model is visualized in Figure 6.

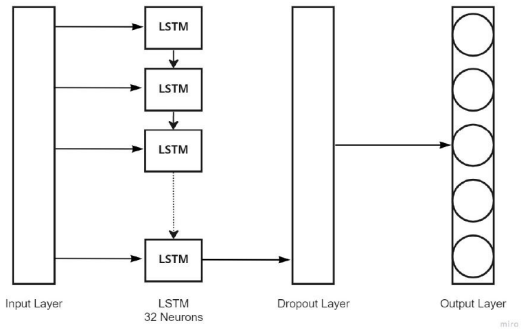


Fig. 6: Single Layer LSTM architecture

Double Layer LSTM Model has two layers of LSTM cells and a dropout layer between them. This time first LSTM layer sends all hidden states to the next layers and another LSTM layer after them. The LSTM layers have tanh activation functions and 32 neurons in their hidden states, and the final dense layer has the number of neurons by output size. The model is visualized in Figure 7.

The bidirectional LSTM (BDLSTM) Model has a single layer with connected bidirectional components which have two LSTM cells in themselves that are connected to each other in forward and backward directions among bidirectional components. Each LSTM cell has again 32 neurons in their hidden states and tanh activation function is used again. The bidirectional layer is followed by a dropout layer and an output layer with the number of neurons by output size. The model is visualized in Figure 8.

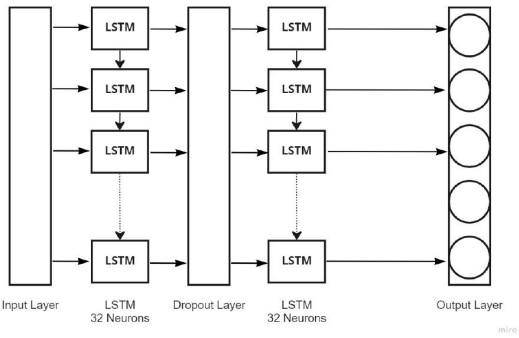


Fig. 7: Double Layer LSTM architecture

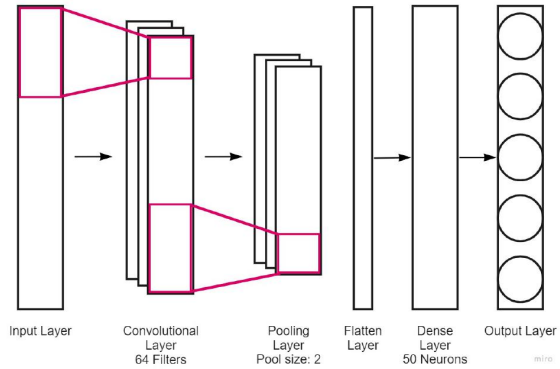


Fig. 9: 1D-CNN architecture

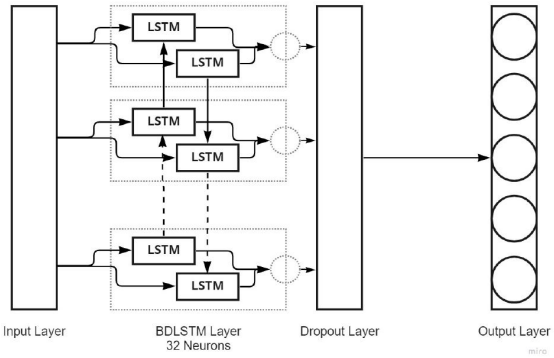


Fig. 8: BDLSTM architecture

The 1D CNN Model has a completely different structure and idea. Such as the regular convolutional operation in image processing, this model has a convolution layer but a 1-dimensional one. The convolutional layer has 64 filters in itself and a pooling layer is following it with a pooling size of 2. A flatten layer converts the pooling output to a 1-dimensional vector and a dense layer with 50 neurons is applied. Finally, the output layer takes the output of the dense layer and produces an output with desired output size. The model is visualized in Figure 9.

C. Datasets

In order to observe different behaviors against adversarial attacks multiple datasets have been gathered and multiple experiments have been conducted. Four different datasets have been gathered and can be summarized as follows:

- Electricity Transformer Data[1]: is a multivariate time series dataset that contains measurements of the amount of electricity used in a single home over a period of almost 4 years (December 2006 to November 2010) in 15 minutes time-intervals. These measurements were taken using sub-meters installed in three different locations. In the experiments, only the electricity data is collected and used for time-series forecasting tasks. (Electricity Transformer)
- Metro Interstate Human Traffic Volume[2]: is a dataset of hourly traffic volume with the weather forecast and holiday information in Interstate 94 Westbound, MN DoT ATR station 301. (Metro Traffic)
- Beijing-Guanyuan Air-Quality[3]: This data set includes hourly air pollutants data with weather information from 12 nationally-controlled air-quality monitoring sites but only Guanjuan region is collected. The air-quality data are from the Beijing Municipal Environmental Monitoring Center. The time period is from March 1st, 2013 to February 28th, 2017. (Air Quality)
- Solar Generation[4]: is dataset gathered from Kaggle and related to a competition organized by EnerjiSA. The dataset has solar generation data with multiple weather condition information recorded between January 1st, 2019, and November 30th, 2021 in the capital district of Turkey contains 7 cities. (Solar Generation)

These datasets have been used for different experiment setups. The prepared inputs from each dataset have been

[1]https://github.com/zhouhaoyi/ETDataset

[2]https://archive.ics.uci.edu/ml/datasets/Metro+Interstate+Traffic+Volume

[3]https://archive.ics.uci.edu/ml/datasets/Beijing+Multi-Site+Air-Quality+Data

[4]https://www.kaggle.com/competitions/enerjisa-enerji-veri-maratonu/data

TABLE I: Input and output lengths by datasets

	Total	Many-to-one		Many-to-many	
		Input	Output	Input	Output
Electricity Transformer	69680	92	1	168	12
Metro Traffic	48204	23	1	168	12
Air-Quality	35064	23	1	168	12
Solar Generation	26304	23	1	168	12

utilized for each model that has been introduced earlier. For many-to-one and many-to-many approaches different sizes of inputs have been selected. The details about input and output sizes are given in Table I.

D. Adversarial Attacks

After each of the models was trained for each dataset, FGSM and PGD attacks have been applied to the samples by using models. Mode et al. [26] used FGSM and BIM methods and for each of the attacks, they have defined $\epsilon = 0.2$. For an FGSM attack, only the epsilon value needs to be selected, we have selected values in $[0.1, 0.05, 0.025, 0.01]$, and they are applied one by one. For PGD, alpha, epsilon and no. iteration values should be selected and alpha was in $[0.025, 0.01]$, epsilon was in $[0.025, 0.01]$ and no. iterations were fixed to 7. For each type of attack, a minimum and maximum range are defined such as $[0, \infty]$ in order to prevent the attack to generate negative values in the samples.

E. Experiment Setup

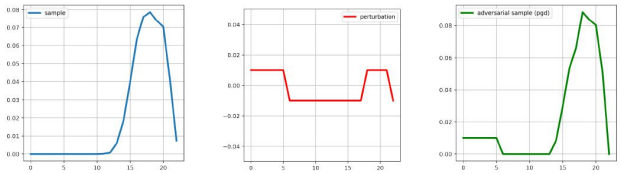
All the experiments were run on Ubuntu 22.04 with AMD Ryzen 9 CPU, 32GB RAM, and NVidia Titan X GPU. The models have been created and adversarial attacks have been applied with Tensorflow.

V. Results

The experiments have been conducted with the settings explained in the previous section. There are 2 different experiment results,

- Forecasting performances of models with adversarial samples for both many-to-one and many-to-many settings.
- Forecasting performances of models after training with adversarial samples.

Before interpreting the experiment results, which are the model performances, it is insightful to observe behaviors of perturbations and adversarial attacks for different datasets. The adversarial attacks have been applied to different datasets because time series patterns and components, such as seasonality, trend, and randomness, differ in different domains, and that shows an effect on the training of the forecasting models. For example, we can make an assumption such as for input with a daily seasonality and hourly interval, the forecasting next hour has a strong relationship with the value of the same hour of yesterday. Therefore, while the model is being trained, the model would give a high gradient value for the time input of one day ago. When it comes to the adversarial attacks, since our adversarial sample generation algorithms, FGSM and PGD, are calculated with gradient, they would generate a perturbation that perturbs the important time inputs. The pattern of the dataset and the adversarial samples that are generated by adversarial sample generators are strongly related.



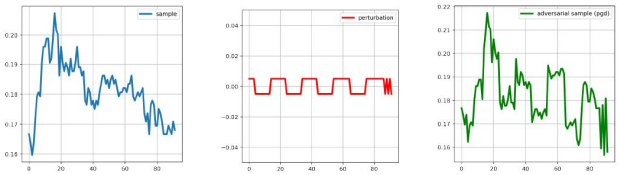
(a) Original Sample (b) Perturbation (c) Adv. Sample

Fig. 10: (Many-to-one) Original sample, perturbation, and Adversarial sample generated with PGD attack and LSTM model for Solar Generation dataset with input size of 23

The Solar Generation dataset consists of the data generated energy from solar panels, therefore in night periods, the energy generation would be zero since there would be no sun. In Figure 10a, there is the sample data from the Solar Generation dataset and in Figure 10b, we can see the perturbation that is generated with LSTM model. The forecasting model takes 23 time inputs and in the perturbation chart, it is clear to see that the last and first values in the time series input were tried to perturbed by the attacks since those time steps have importance in the forecast.

When we tried to observe the effects of the attacks for a longer time series sequence, Electricity Transformer data was investigated with 92 time steps in its input. That time in Figure 11a, it is clear to see that there is a long sequence of input with a different pattern. In Figure 11b, the perturbation has a different pattern and an oscillation-like behavior for the last time steps. There are multiple perturbed regions in the data which can be interpreted by Figure 11b and seen in Figure 11c.

In Figure 12, the original input and adversarial samples generated with FGSM and PGD have been compared. The FGSM and PGD have generated similar



(a) Original Sample (b) Perturbation (c) Adv. Sample

Fig. 11: (Many-to-one) Original sample, perturbation, and Adversarial sample generated with PGD attack and LSTM model for Electricity Transformer dataset with input size of 92

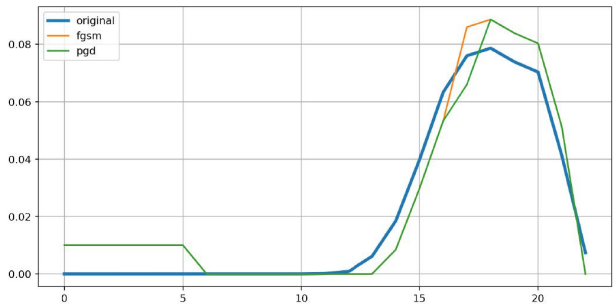


Fig. 12: (Many-to-one) Comparison of Adversarial Samples generated with FGSM and PGD attacks and LSTM model for Solar Generation dataset with input size of 23

adversarial attacks but they are just different for a couple of time steps, for the given figure time steps in the peak values.

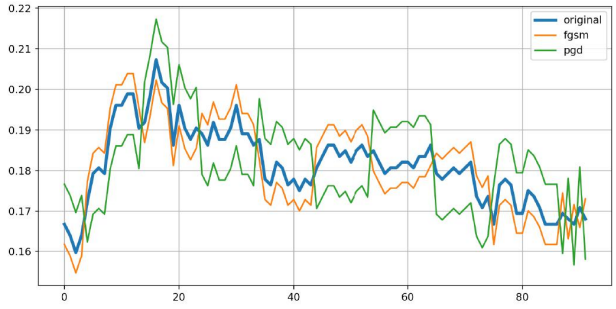
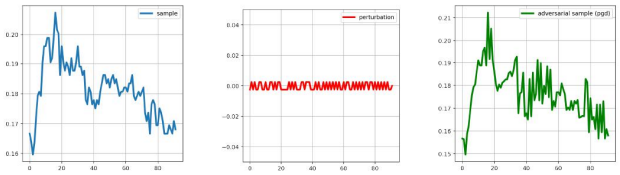


Fig. 13: (Many-to-one) Comparison of Adversarial Samples generated with FGSM and PGD attacks and LSTM model for Electricity Transformer dataset with input size of 92

The same comparison for the Electricity Transformer dataset was conducted and can be seen in Figure 13. This time the behaviors of the adversarial samples generated by FGSM and PGD are different. For multiple regions in the sample input, they have perturbed the original sample in reverse order. In other words, when FGSM generated a positive perturbation value, PGD generated a negative value, and vice versa.



(a) Original Sample (b) Perturbation (c) Adv. Sample

Fig. 14: (Many-to-one) Original sample, perturbation, and Adversarial sample generated with PGD attack and CNN model for Electricity Transformer dataset with input size of 92

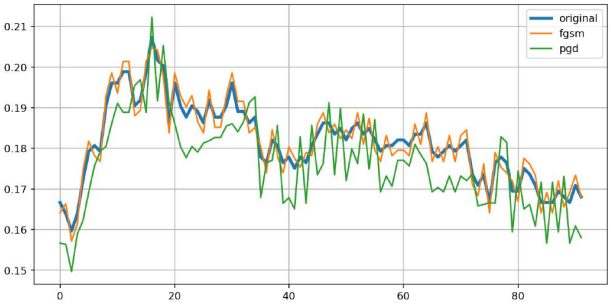
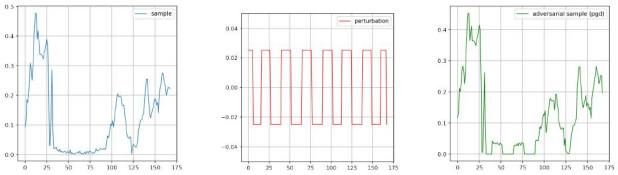


Fig. 15: (Many-to-one) Comparison of Adversarial Samples generated with FGSM and PGD attacks and CNN model for Electricity Transformer dataset with input size of 92

Since the adversarial sample generation algorithms use gradients and different forecasting models would have different algorithms and gradient calculations with respect to that, the generated perturbations and the adversarial sample would differ. In Figure 14 and Figure 15, we have used Electricity Transformer for same time steps in input but this time 1D-CNN model has been utilized. Due to gradient values calculated by CNN, the perturbation generated by PGD attack has a completely different pattern when compared with the LSTM. The perturbed samples behave as they have an oscillation, especially for the sample generated by the FGSM. We could observe that, the used deep learning model has an important effect on generated adversarial samples.



(a) Original Sample (b) Perturbation (c) Adv. Sample

Fig. 16: (Many-to-many) Original sample, perturbation, and Adversarial sample generated with PGD attack and LSTM model for Electricity Transformer dataset with input size of 168

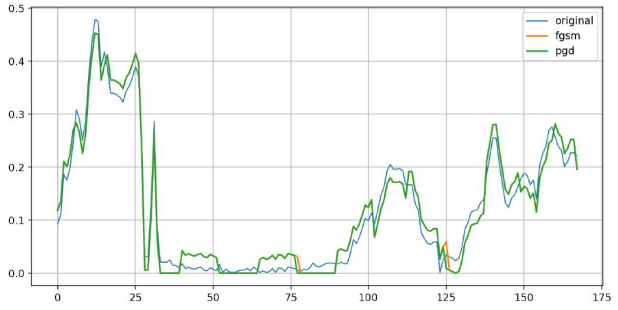


Fig. 17: (Many-to-many) Comparison of Adversarial Samples generated with FGSM and PGD attacks and LSTM model for Electricity Transformer dataset with input size of 168

As mentioned above, there are two strategies when forecasting the future; one is predicting the next time step and the other is predicting a sequence of time steps. In our case, that property defines the output size of the forecasting models. The observations that are mentioned before have been conducted with a many-to-one strategy. Since the output size would be decisive in calculation of gradients, we have used a many-to-many strategy to observe the behavior of the adversarial samples. In the Figure 16a, we have shown the original sample of the Electricity Transformer dataset for an input size of 168 time steps. In Figure 16b, there is the perturbation generated by PGD attack with LSTM model. The attacks have similar perturbations in multiple regions and have a similar pattern with the perturbation generated with many-to-one strategy (Figure 14b). This time FGSM and PGD algorithms generated very close adversarial samples, that can be seen in Figure 17.

A. Adversarial Attacks to Forecasting Models

The adversarial samples have been generated with FGSM and PGD algorithms in different settings described in the previous section. Adversarial samples are generated and given to the trained models as inputs in order the observe the forecasting performances in four different forecasting error metrics (R^2, RMSE, MAE, MAPE). The forecasting performances in many-to-one strategy for both original samples and adversarial samples have been given in Appendix A-A but for only MAE metric the results have been given in Table II.

For different datasets, the models with the best performance have been changed in different adversarial sample generation settings. Overall, it can be said that the samples generated with the FGSM algorithm are slightly better at failing the DL models in forecasting tasks. For the Electricity Transformer dataset, the Single Layer LSTM and BDLSTM models have performed the best forecasting performance against adversarial samples, which is similar to the Metro Traffic and Air-Quality datasets. Interestingly, for the Solar Generation dataset, Double Layer LSTM and CNN1D models performed better than others.

The experiences on many-to-many strategy have been conducted as well. The experiment results for different error metrics have been given in Appendix A-B and for only MAE metric they are given in Table III. In the many-to-many strategy, Single Layer LSTM and Double Layer LSTM models have shown the best performances for multiple datasets but for the Air-Quality dataset, BDLSTM dominated all other models in terms of forecasting performance, in another perspective robustness against adversarial samples.

B. Adversarial Training

Adversarial training is a technique used in machine learning to improve the robustness of a model by exposing it to adversarial examples during training. The idea behind adversarial training is to use these adversarial examples as additional training data so that the model learns to be more robust to such perturbations. This can be thought of as a form of regularization, as it helps the model generalize better to inputs that are slightly different from those it was trained on.

In order to generate adversarial training samples for the forecasting models, we have used both FGSM and PGD algorithms with $\alpha = 0.025$ and $\epsilon = 0.025$, since in the previous experiments we observed that those settings are enough to fail forecasting models. Generated adversarial samples added to the both training and test

TABLE II: Many-to-One - Forecasting Performances (MAE Score) of Adversarial Samples on Models

		Original	FGSM (ϵ)				PGD (α/ϵ)			
			0.1	**0.05**	**0.025**	**0.01**	**0.025 / 0.01**	**0.025 / 0.025**	**0.01 / 0.01**	**0.01 / 0.025**
Electricity Transformer	**Single Layer LSTM**	*0,25*	6,05	3,16	1,7	0,83	**0,56**	**1,32**	**0,56**	**1,32**
	Double Layer LSTM	***0,22***	5,58	2,93	1,58	0,76	0,62	1,38	0,62	1,38
	BDLSTM	*0,23*	**5,51**	**2,89**	**1,56**	**0,76**	0,59	1,32	0,59	1,32
	CNN1D	*0,39*	6,26	3,24	1,94	1,07	0,89	2,34	1,1	2,54
Metro Traffic	**Single Layer LSTM**	*321,11*	1671,08	**1081,7**	**710,75**	477,62	473,22	**684,65**	473,33	**683,92**
	Double Layer LSTM	*306,83*	1839,13	1272,24	825,71	515,52	511,21	805,16	511,73	805,67
	BDLSTM	***305,38***	**1637,23**	1106,04	723,97	**474,39**	**469,45**	697,62	**469,69**	696,92
	CNN1D	*325,54*	1388,14	1049,14	771,21	533,99	525,7	801,82	542,75	827,78
Air-Quality	**Single Layer LSTM**	***10,58***	**82,29**	**49,28**	**31,01**	**19,15**	17,83	27,41	17,87	27,42
	Double Layer LSTM	*10,77*	89,6	52,16	32,09	19,5	17,37	25,83	17,52	25,9
	BDLSTM	*10,79*	87,66	50,84	31,33	19,16	**17,08**	**24,48**	**17,12**	**24,52**
	CNN1D	*12,59*	116	69,01	42,28	24,98	23,31	36,91	23,98	37,18
Solar Generation	**Single Layer LSTM**	*8,16*	98,27	56,45	33,14	18,38	18,21	32,54	18,24	32,95
	Double Layer LSTM	***6,12***	74,44	45,61	**27,58**	**15,19**	**15,05**	28,08	**15,11**	**27,94**
	BDLSTM	*8,94*	115,69	70,71	42,34	22,99	21,57	39,5	21,75	40,05
	CNN1D	*7,11*	**64,96**	**42,92**	28,7	16,93	16,44	**28,05**	16,78	30,65

set while considering those sets homogeneous in terms of original and adversarial examples. The error metrics for original samples with the original model, error metrics of attacking the original model, and error metrics for training and testing test set after training with adversarial samples are given in Appendix B, but for only MAE metric results have been given in Table IV.

Adversarial training showed an important decrease in error metrics and therefore increase in the forecasting performance for each of the models. Even for the Solar Generation dataset, the BDLSTM model performed better than the performance of the original model, the adversarial training can be useful for generalization.

VI. Conclusion and Future Work

In that study, we experienced the effects of adversarial attacks on time series forecasting tasks. Two different adversarial sample generation algorithms have been used; FGSM and PGD. Adversarial attacks have been applied to four different forecasting models, which are LSTM and CNN-based models, for four different datasets, where each one has different characteristics in terms of time series data. For the many-to-one strategy, the BDLSTM model has shown the best performance while Single and Double Layer LSTM models have shown the best performance in the many-to-many strategy. To build robust forecasting models against adversarial attacks, we have trained forecasting models with original and adversarial samples and that technique has shown a strong increase in the forecasting performance, therefore the adversarial training technique can be used for training forecasting models.

In future work, the effect of adversarial attacks in multivariate time series forecasting tasks can be investigated. There are multiple works that studied multivariate time series forecasting tasks and their robustness can be different because of the dimensionality.

References

[1] G. Montavon, W. Samek, and K.-R. Müller, "Methods for interpreting and understanding deep neural networks," *Digital signal processing*, vol. 73, pp. 1–15, 2018.

[2] S. Bhanja and A. Das, "Deep neural network for multivariate time-series forecasting," in *Proceedings of international conference on frontiers in computing and systems*, pp. 267–277, Springer, 2021.

[3] S. Selvin, R. Vinayakumar, E. Gopalakrishnan, V. K. Menon, and K. Soman, "Stock price prediction using lstm, rnn and cnn-sliding window model," in *2017 international conference on advances in computing, communications and informatics (icacci)*, pp. 1643–1647, IEEE, 2017.

[4] H. Jahangir, H. Tayarani, S. S. Gougheri, M. A. Golkar, A. Ahmadian, and A. Elkamel, "Deep learning-based forecasting approach in smart grids with microclustering and bidirectional lstm network," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 9, pp. 8298–8309, 2020.

TABLE III: Many-to-Many - Forecasting Performances (MAE Score) of Adversarial Samples on Models

		Original	FGSM (ϵ)				PGD (α/ϵ)			
			0.1	**0.05**	**0.025**	**0.01**	**0.025 / 0.01**	**0.025 / 0.025**	**0.01 / 0.01**	**0.01 / 0.025**
Electricity Transformer	**Single Layer LSTM**	*0,75*	5,64	3,17	**1,94**	1,2	1,09	**1,11**	1,09	**1,1**
	Double Layer LSTM	*0,56*	**6**	**3,38**	1,95	1,08	1	1,67	**1**	1,67
	BDLSTM	***0,54***	6,36	3,43	1,95	**1,07**	**0,95**	1,61	0,95	1,61
	CNN1D	*0,64*	8,69	4,76	2,89	1,72	1,21	3,02	1,76	3,89
Metro Traffic	**Single Layer LSTM**	*883,58*	1992,85	1666,14	1362,27	1096,78	1092,29	1410,13	1105,81	1415,18
	Double Layer LSTM	***747,33***	**1864,41**	1567,2	**1250,9**	**961,99**	**965,24**	**1322,48**	**972,16**	**1319,71**
	BDLSTM	*848,21*	1902,08	**1556,95**	1268,15	1034,15	1033,8	1324,13	1045,96	1323,33
	CNN1D	*1007,95*	2335,42	2072,83	1718,84	1354,73	1332,08	1834,8	1381,13	1901,25
Air-Quality	**Single Layer LSTM**	*33,09*	102,25	68,24	50,25	39,52	39,28	48,2	39,29	48,19
	Double Layer LSTM	*33,22*	103,18	69,98	51,42	40,12	40,01	50,19	40,02	50,15
	BDLSTM	***33,03***	**91,41**	**62,5**	**47,3**	**38,34**	**38,15**	**45,74**	**38,15**	**45,65**
	CNN1D	*39,54*	139,28	94,64	70,76	54,94	51,17	76,96	56,39	85,01
Solar Generation	**Single Layer LSTM**	*26,28*	**66,68**	**46,44**	**36,01**	29,98	29,99	**36**	29,99	35, 9
	Double Layer LSTM	*23,99*	79,71	52,32	37,73	**29,18**	**29,16**	37,44	**29,17**	37,45
	BDLSTM	*26,77*	83,57	54,72	40,15	31,8	31,83	40,43	31,83	40,26
	CNN1D	***22,27***	81,84	61,73	46,36	32,8	32,4	47,85	32,94	51,53

[5] I. Goodfellow, P. McDaniel, and N. Papernot, "Making machine learning robust against adversarial inputs," *Communications of the ACM*, vol. 61, no. 7, pp. 56–66, 2018.

[6] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *arXiv preprint arXiv:1312.6199*, 2013.

[7] S. Huang, N. Papernot, I. Goodfellow, Y. Duan, and P. Abbeel, "Adversarial attacks on neural network policies," *arXiv preprint arXiv:1702.02284*, 2017.

[8] Y. Cao, C. Xiao, B. Cyr, Y. Zhou, W. Park, S. Rampazzi, Q. A. Chen, K. Fu, and Z. M. Mao, "Adversarial sensor attack on lidar-based perception in autonomous driving," in *Proceedings of the 2019 ACM SIGSAC conference on computer and communications security*, pp. 2267–2281, 2019.

[9] Y. Deng, X. Zheng, T. Zhang, C. Chen, G. Lou, and M. Kim, "An analysis of adversarial attacks and defenses on autonomous driving models," in *2020 IEEE international conference on pervasive computing and communications (PerCom)*, pp. 1–10, IEEE, 2020.

[10] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[11] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," *arXiv preprint arXiv:1706.06083*, 2017.

[12] S. L. Ho and M. Xie, "The use of arima models for reliability forecasting and analysis," *Computers & industrial engineering*, vol. 35, no. 1-2, pp. 213–216, 1998.

[13] S. Siami-Namini, N. Tavakoli, and A. S. Namin, "A comparison of arima and lstm in forecasting time series," in *2018 17th IEEE international conference on machine learning and applications (ICMLA)*, pp. 1394–1401, IEEE, 2018.

[14] F.-M. Tseng and G.-H. Tzeng, "A fuzzy seasonal arima model for forecasting," *Fuzzy Sets and Systems*, vol. 126, no. 3, pp. 367–376, 2002.

[15] H. Chen, C. A. Canizares, and A. Singh, "Ann-based short-term load forecasting in electricity markets," in *2001 IEEE power engineering society winter meeting. Conference proceedings (Cat. No. 01CH37194)*, vol. 2, pp. 411–415, IEEE, 2001.

[16] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, "Short-term residential load forecasting based on lstm recurrent neural network," *IEEE Transactions on Smart Grid*, vol. 10, no. 1, pp. 841–851, 2017.

[17] G. P. Zhang, "Time series forecasting using a hybrid arima and neural network model," *Neurocomputing*, vol. 50, pp. 159–175, 2003.

[18] V. K. R. Chimmula and L. Zhang, "Time series forecasting of covid-19 transmission in canada using lstm networks," *Chaos, Solitons & Fractals*, vol. 135, p. 109864, 2020.

[19] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[20] K. Jaseena and B. C. Kovoor, "Decomposition-based hybrid wind speed forecasting model using deep bidirectional lstm networks," *Energy Conversion and Management*, vol. 234, p. 113944, 2021.

[21] I. E. Livieris, E. Pintelas, and P. Pintelas, "A cnn–lstm model for gold price time-series forecasting," *Neural computing and applications*, vol. 32, no. 23, pp. 17351–17360, 2020.

[22] F. Karim, S. Majumdar, and H. Darabi, "Adversarial attacks on time series," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 10, pp. 3309–3320, 2020.

[23] H. I. Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Muller, "Adversarial attacks on deep neural networks for time series classification," in *2019 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, IEEE, 2019.

[24] M. G. Abdu-Aguye, W. Gomaa, Y. Makihara, and Y. Yagi, "Detecting adversarial attacks in time-series data," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech*

TABLE IV: Adversarial Training Performances (MAE Metric)

		Original	FGSM			PGD		
			Attack	**Training**	**Test**	**Attack**	**Training**	**Test**
Electricity Transformer	**Single Layer LSTM**	*0,25*	1,7	0,38	0,42	1,32	0,39	0,34
	Double Layer LSTM	*0,22*	1,58	0,4	0,24	1,38	0,47	0,38
	Bidirectional LSTM	*0,23*	1,56	0,33	0,27	1,32	0,44	0,4
	CNN1D	*0,39*	1,94	0,61	0,41	2,34	0,61	0,37
Metro Traffic	**Single Layer LSTM**	*321,11*	710,75	389,55	314,98	684,65	416,73	373,3
	Double Layer LSTM	*306,83*	825,71	346,07	281,6	805,16	351,6	304,21
	Bidirectional LSTM	*305,38*	723,97	380,06	313,59	697,62	370,6	311,65
	CNN1D	*325,54*	771,21	361,46	324,38	801,82	356,77	317,47
Air-Quality	**Single Layer LSTM**	*10,58*	31,01	13,23	14,07	27,41	13,02	13,59
	Double Layer LSTM	*10,77*	32,09	12,7	12,47	25,83	12,6	13,26
	Bidirectional LSTM	*10,79*	31,33	11,81	11,56	24,48	12,28	12,69
	CNN1D	*12,59*	42,28	13,09	13,98	36,91	11,86	12,41
Solar Generation	**Single Layer LSTM**	*8,16*	33,14	7,36	8,52	32,54	7,66	9,88
	Double Layer LSTM	*6,12*	27,58	5,42	6,27	28,08	5,61	6,66
	Bidirectional LSTM	*8,94*	42,34	6,62	7,59	39,5	7,48	9,26
	CNN1D	*7,11*	28,7	7,91	9,33	28,05	6,71	7,93

and Signal Processing (ICASSP), pp. 3092–3096, IEEE, 2020.

[25] T. Wu, X. Wang, S. Qiao, X. Xian, Y. Liu, and L. Zhang, "Small perturbations are enough: Adversarial attacks on time series prediction," *Information Sciences*, vol. 587, pp. 794–812, 2022.

[26] G. R. Mode and K. A. Hoque, "Adversarial examples in deep learning for multivariate time series regression," in *2020 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, pp. 1–10, IEEE, 2020.

[27] Z. Cui, R. Ke, Z. Pu, and Y. Wang, "Deep bidirectional and unidirectional lstm recurrent neural network for network-wide traffic speed prediction," *arXiv preprint arXiv:1801.02143*, 2018.

[28] E. Chaerun Nisa and Y.-D. Kuan, "Comparative assessment to predict and forecast water-cooled chiller power consumption using machine learning and deep learning algorithms," *Sustainability*, vol. 13, no. 2, p. 744, 2021.

APPENDIX A
FORECASTING PERFORMANCES ON ADVERSARIAL SAMPLES OF MODELS

A. *Many-to-one*

TABLE V: Many-to-One - Forecasting Performances on Adversarial Samples of Models

			Original	FGSM (ϵ)				PGD (α/ϵ)			
				0.1	0.05	0.025	0.01	0.025 / 0.01	0.025 / 0.025	0.01 / 0.01	0.01 / 0.025
Electricity Transformer	Single Layer LSTM	R2	0,99	-1,737	0,252	0,779	0,944	0,967	0,863	0,967	0,863
		RMSE	0,35	6,06	3,17	1,72	0,87	0,66	1,35	0,66	1,35
		MAE	0,25	6,05	3,16	1,7	0,83	0,56	1,32	0,56	1,32
		MAPE	4,94	116,14	60,39	32,6	15,99	10,62	25,08	10,61	25,08
	Double Layer LSTM	R2	0,99	-1,334	0,354	0,81	0,952	0,964	0,85	0,964	0,85
		RMSE	0,32	5,6	2,94	1,6	0,8	0,7	1,42	0,7	1,42
		MAE	0,22	5,58	2,93	1,58	0,76	0,62	1,38	0,62	1,38
		MAPE	4,34	108,46	56,01	30,11	14,65	12,15	26,72	12,15	26,72
	BDLSTM	R2	0,99	-1,274	0,373	0,815	0,953	0,966	0,861	0,966	0,861
		RMSE	0,32	5,52	2,9	1,58	0,79	0,67	1,37	0,67	1,37
		MAE	0,23	5,51	2,89	1,56	0,76	0,59	1,32	0,59	1,32
		MAPE	4,32	105,03	54,66	29,47	14,37	10,8	24,69	10,8	24,69
	CNN1D	R2	0,98	-2,379	0,114	0,686	0,899	0,923	0,566	0,891	0,492
		RMSE	0,53	6,73	3,45	2,05	1,16	1,02	2,41	1,21	2,61
		MAE	0,39	6,26	3,24	1,94	1,07	0,89	2,34	1,1	2,54
		MAPE	7,8	119,37	60,98	36,56	20,66	17,27	44,37	20,87	47,69
Metro Traffic	Single Layer LSTM	R2	0,95	0,174	0,635	0,831	0,912	0,912	0,835	0,912	0,836
		RMSE	449,74	1795,6	1192,95	812,3	585,76	584,75	801,47	584,83	800,48
		MAE	321,11	1671,08	1081,7	710,75	477,62	473,22	684,65	473,33	683,92
		MAPE	17,21	129,43	79,73	48,67	29,64	29,02	44,42	29,04	44,59
	Double Layer LSTM	R2	0,95	-0,147	0,441	0,76	0,898	0,899	0,761	0,898	0,761
		RMSE	433,64	2115,27	1477,42	968,36	629,55	628,99	966,2	629,4	966,33
		MAE	306,83	1839,13	1272,24	825,71	515,52	511,21	805,16	511,73	805,67
		MAPE	19,19	181,15	116,04	67,41	37,49	36,93	62,76	36,96	62,94
	BDLSTM	R2	0,95	0,181	0,615	0,826	0,915	0,915	0,831	0,915	0,831
		RMSE	429,81	1787,95	1226,29	823,86	576,94	575,66	813,06	575,83	812,16
		MAE	305,38	1637,23	1106,04	723,97	474,39	469,45	697,62	469,69	696,92
		MAPE	15,65	123,28	81,14	49,49	29,14	28,42	45,47	28,46	45,8
	CNN1D	R2	0,95	0,336	0,637	0,804	0,898	0,899	0,79	0,895	0,779
		RMSE	448,97	1609,77	1190,17	874,36	631,58	626,65	904,74	640,99	927,69
		MAE	325,54	1388,14	1049,14	771,21	533,99	525,7	801,82	542,75	827,78
		MAPE	19,32	73,57	62,34	48,05	33,21	32,65	49,4	33,95	52,25
Air-Quality	Single Layer LSTM	R2	0,95	-0,052	0,618	0,833	0,917	0,92	0,852	0,92	0,852
		RMSE	19,57	91,29	55,02	36,36	25,66	25,22	34,26	25,24	34,26
		MAE	10,58	82,29	49,28	31,01	19,15	17,83	27,41	17,87	27,42
		MAPE	26,98	240,88	143,83	92	56,71	47,74	68,11	48,11	68,16
	Double Layer LSTM	R2	0,95	-0,155	0,595	0,827	0,915	0,92	0,858	0,92	0,858
		RMSE	19,67	95,66	56,68	37	25,88	25,19	33,51	25,24	33,54
		MAE	10,77	89,6	52,16	32,09	19,5	17,37	25,83	17,52	25,9
		MAPE	33,26	318,95	182,15	110,98	65,91	53,35	75,39	54,76	76,33
	BDLSTM	R2	0,95	-0,107	0,611	0,833	0,917	0,921	0,866	0,921	0,866
		RMSE	19,69	93,67	55,49	36,38	25,67	25,02	32,61	25,03	32,62
		MAE	10,79	87,66	50,84	31,33	19,16	17,08	24,48	17,12	24,52
		MAPE	34,46	314,26	178,14	108,49	65,13	54,27	72,71	54,71	73,21
	CNN1D	R2	0,94	-0,909	0,318	0,724	0,879	0,885	0,759	0,881	0,755
		RMSE	21,56	123,01	73,51	46,78	30,96	30,17	43,7	30,72	44,03
		MAE	12,59	116	69,01	42,28	24,98	23,31	36,91	23,98	37,18
		MAPE	46,36	449,96	253,91	152,31	88,6	85,39	139,22	86,98	137,73
Solar Generation	Single Layer LSTM	R2	0,99	0,216	0,724	0,901	0,968	0,968	0,904	0,968	0,903
		RMSE	12,86	121,95	72,39	43,31	24,79	24,67	42,69	24,7	42,98
		MAE	8,16	98,27	56,45	33,14	18,38	18,21	32,54	18,24	32,95
		MAPE	895897,9	18428799	8460727	4382258	2228543	2307727	4793283	2308098	4794612
	Double Layer LSTM	R2	0,99	0,56	0,83	0,935	0,978	0,978	0,937	0,978	0,938
		RMSE	10,7	91,33	56,82	35,21	20,51	20,25	34,5	20,29	34,41
		MAE	6,12	74,44	45,61	27,58	15,19	15,05	28,08	15,11	27,94
		MAPE	714818,6	13832860	7439727	4137325	2105847	2579360	6136413	2579907	6136541
	BDLSTM	R2	0,99	0,169	0,682	0,883	0,963	0,966	0,896	0,966	0,893
		RMSE	13,06	125,56	77,72	47,15	26,52	25,44	44,36	25,58	45,04
		MAE	8,94	115,69	70,71	42,34	22,99	21,57	39,5	21,75	40,05
		MAPE	1550266	38006422	22815919	13121066	6472724	5801118	11956808	6209133	12718413
	CNN1D	R2	0,99	0,647	0,84	0,925	0,972	0,972	0,926	0,972	0,916
		RMSE	12,23	81,8	55,09	37,83	23,25	22,86	37,46	23,21	39,87
		MAE	7,11	64,96	42,92	28,7	16,93	16,44	28,05	16,78	30,65
		MAPE	575486,5	12133303	6893407	3423600	1685250	1496755	3669001	1547500	3751907

B. Many-to-many

TABLE VI: Many-to-Many - Forecasting Performances on Adversarial Samples of Models

			Original	FGSM				PGD			
				0.1	0.05	0.025	0.01	0.025 / 0.01	0.025 / 0.025	0.01 / 0.01	0.01 / 0.025
Electricity Transformer	Single Layer LSTM	R2	0,93	-1,403	0,218	0,686	0,859	0,869	0,844	0,869	0,844
		RMSE	0,98	5,68	3,24	2,05	1,38	1,33	1,45	1,33	1,45
		MAE	0,75	5,64	3,17	1,94	1,2	1,09	1,11	1,09	1,1
		MAPE	14,92	107,92	60,92	37,42	23,47	21,39	20,43	21,39	20,39
	Double Layer LSTM	R2	0,95	-1,782	0,096	0,677	0,88	0,889	0,745	0,889	0,747
		RMSE	0,81	6,11	3,48	2,08	1,27	1,22	1,85	1,22	1,84
		MAE	0,56	6	3,38	1,95	1,08	1	1,67	1	1,67
		MAPE	10,17	116,1	64,17	36,55	19,84	17,89	30,56	17,86	30,36
	BDLSTM	R2	0,95	-2,061	0,087	0,683	0,884	0,896	0,762	0,896	0,764
		RMSE	0,8	6,41	3,5	2,06	1,25	1,18	1,79	1,18	1,78
		MAE	0,54	6,36	3,43	1,95	1,07	0,95	1,61	0,95	1,61
		MAPE	10,32	122,8	65,65	37,14	20,21	17,78	30,08	17,77	29,97
	CNN1D	R2	0,94	-5,761	-1,052	0,253	0,726	0,83	0,177	0,703	-0,272
		RMSE	0,92	9,53	5,25	3,17	1,92	1,51	3,32	2	4,13
		MAE	0,64	8,69	4,76	2,89	1,72	1,21	3,02	1,76	3,89
		MAPE	12,43	161,58	90,8	55,9	33,16	24,6	55,93	34,07	71,82
Metro Traffic	Single Layer LSTM	R2	0,63	-0,476	-0,076	0,246	0,476	0,48	0,193	0,469	0,188
		RMSE	1195,59	2398,26	2047,63	1714,56	1429,15	1423,8	1773,71	1439,1	1779,46
		MAE	883,58	1992,85	1666,14	1362,27	1096,78	1092,29	1410,13	1105,81	1415,18
		MAPE	68,7	155,89	126,55	102,96	83,88	83,62	107,98	84,58	108,58
	Double Layer LSTM	R2	0,7	-0,397	-0,048	0,286	0,54	0,538	0,217	0,532	0,218
		RMSE	1090	2333,5	2020,92	1668,47	1339,71	1342,64	1747,49	1350,85	1746,09
		MAE	747,33	1864,41	1567,2	1250,9	961,99	965,24	1322,48	972,16	1319,71
		MAPE	56,82	162,02	127,59	98,01	73,94	74,52	105,31	74,93	104,76
	BDLSTM	R2	0,65	-0,383	0,033	0,318	0,512	0,512	0,264	0,502	0,262
		RMSE	1167,69	2321,82	1941,77	1630,2	1378,67	1378,92	1693,36	1393,53	1696,06
		MAE	848,21	1902,08	1556,95	1268,15	1034,15	1033,8	1324,13	1045,96	1323,33
		MAPE	70,76	165,67	129,96	105,01	86,13	86,36	111,34	87,25	111,71
	CNN1D	R2	0,55	-1,056	-0,564	-0,1	0,274	0,298	-0,213	0,254	-0,297
		RMSE	1317,6	2830,58	2468,96	2071,08	1682,23	1654,5	2174,41	1705,36	2248,59
		MAE	1007,95	2335,42	2072,83	1718,84	1354,73	1332,08	1834,8	1381,13	1901,25
		MAPE	77,23	215,73	158,26	125,51	99,9	97,83	132,17	101,18	136,73
Air-Quality	Single Layer LSTM	R2	0,63	-0,669	0,131	0,424	0,558	0,559	0,437	0,559	0,437
		RMSE	54,27	115,1	83,05	67,64	59,22	59,17	66,88	59,17	66,87
		MAE	33,09	102,25	68,24	50,25	39,52	39,28	48,2	39,29	48,19
		MAPE	122,09	408,25	268,63	195,94	151,12	149,43	185,24	149,5	185,19
	Double Layer LSTM	R2	0,62	-0,807	0,042	0,378	0,539	0,539	0,385	0,539	0,386
		RMSE	54,57	119,77	87,21	70,29	60,53	60,5	69,86	60,5	69,83
		MAE	33,22	103,18	69,98	51,42	40,12	40,01	50,19	40,02	50,15
		MAPE	118,24	410,79	267,01	191,91	146,75	145,89	185,81	145,91	185,63
	BDLSTM	R2	0,63	-0,4	0,225	0,457	0,566	0,567	0,466	0,567	0,467
		RMSE	54,54	105,42	78,44	65,65	58,68	58,63	65,1	58,63	65,07
		MAE	33,03	91,41	62,5	47,3	38,34	38,15	45,74	38,15	45,65
		MAPE	121,03	362,89	244,38	182,69	145,05	143,78	174,94	143,79	174,64
	CNN1D	R2	0,51	-2,6	-0,682	-0,041	0,271	0,337	-0,181	0,248	-0,383
		RMSE	62,12	169,07	115,56	90,91	76,08	72,57	96,81	77,27	104,78
		MAE	39,54	139,28	94,64	70,76	54,94	51,17	76,96	56,39	85,01
		MAPE	152,3	671,59	409,1	283,69	212,27	203,26	310,67	217,78	341,9
Solar Generation	Single Layer LSTM	R2	0,9	0,525	0,745	0,834	0,878	0,878	0,834	0,878	0,834
		RMSE	43,05	95,12	69,7	56,18	48,18	48,19	56,28	48,19	56,19
		MAE	26,28	66,68	46,44	36,01	29,98	29,99	36	29,99	35,9
		MAPE	3225615	9306202	4344838	3294525	3179487	3133671	3150965	3135152	3235771
	Double Layer LSTM	R2	0,91	0,324	0,681	0,818	0,879	0,879	0,818	0,879	0,818
		RMSE	41,47	113,44	77,89	58,92	48,05	48,06	58,8	48,07	58,83
		MAE	23,99	79,71	52,32	37,73	29,18	29,16	37,44	29,17	37,45
		MAPE	1723491	10522481	3956065	2118738	1762693	1771927	2153224	1768402	2175797
	BDLSTM	R2	0,9	0,389	0,71	0,825	0,877	0,877	0,824	0,877	0,824
		RMSE	42,57	107,87	74,26	57,75	48,39	48,41	57,94	48,41	57,81
		MAE	26,77	83,57	54,72	40,15	31,8	31,83	40,43	31,83	40,26
		MAPE	4479519	22592772	12704639	8041387	5678131	5700775	8540469	5698980	8561543
	CNN1D	R2	0,92	0,25	0,569	0,745	0,856	0,859	0,734	0,856	0,701
		RMSE	39,74	119,53	90,54	69,69	52,34	51,77	71,13	52,41	75,41
		MAE	22,27	81,84	61,73	46,36	32,8	32,4	47,85	32,94	51,53
		MAPE	1830685	10947467	6156005	3709730	2402870	2368615	3923876	2424367	4950184

APPENDIX B
ADVERSARIAL TRAINING PERFORMANCES

TABLE VII: Adversarial Training Performances

			Original	FGSM			PGD		
				Attack	**Training**	**Test**	**Attack**	**Training**	**Test**
Electricity Transformer	**Single Layer LSTM**	**R2**	*0,99*	0,77	0,99	0,98	0,86	0,99	0,98
		RMSE	*0,35*	1,72	0,53	0,49	1,35	0,55	0,43
		MAE	*0,25*	1,7	0,38	0,42	1,32	0,39	0,34
		MAPE	*4,94*	32,6	3,22	8,47	25,08	3,34	6,96
	Double Layer LSTM	**R2**	*0,99*	0,81	0,99	0,99	0,85	0,99	0,98
		RMSE	*0,32*	1,6	0,57	0,34	1,42	0,64	0,48
		MAE	*0,22*	1,58	0,4	0,24	1,38	0,47	0,38
		MAPE	*4,34*	30,11	3,28	4,6	26,72	4,14	7,47
	Bidirectional LSTM	**R2**	*0,99*	0,81	0,99	0,99	0,86	0,99	0,98
		RMSE	*0,32*	1,58	0,49	0,35	1,37	0,61	0,5
		MAE	*0,23*	1,56	0,33	0,27	1,32	0,44	0,4
		MAPE	*4,32*	29,47	2,8	5,52	24,69	3,73	8,08
	CNN1D	**R2**	*0,98*	0,68	0,98	0,97	0,56	0,99	0,98
		RMSE	*0,53*	2,05	0,87	0,56	2,41	0,83	0,49
		MAE	*0,39*	1,94	0,61	0,41	2,34	0,61	0,37
		MAPE	*7,8*	36,56	4,79	7,58	44,37	4,57	7,27
Metro Traffic	**Single Layer LSTM**	**R2**	*0,95*	0,83	0,9	0,94	0,83	0,9	0,93
		RMSE	*449,74*	812,3	598,86	445,5	801,47	614,81	510,92
		MAE	*321,11*	710,75	389,55	314,98	684,65	416,73	373,3
		MAPE	*17,21*	48,67	37,99	17,4	44,42	58,01	30,25
	Double Layer LSTM	**R2**	*0,95*	0,76	0,92	0,95	0,76	0,92	0,95
		RMSE	*433,64*	968,36	545,45	409,75	966,2	545,06	422,79
		MAE	*306,83*	825,71	346,07	281,6	805,16	351,6	304,21
		MAPE	*19,19*	67,41	37,27	14,23	62,76	45	17,89
	Bidirectional LSTM	**R2**	*0,95*	0,82	0,91	0,95	0,83	0,91	0,95
		RMSE	*429,81*	823,86	578,74	433,3	813,06	574,7	436,6
		MAE	*305,38*	723,97	380,06	313,59	697,62	370,6	311,65
		MAPE	*15,65*	49,49	49,51	19,16	45,47	47,22	18,05
	CNN1D	**R2**	*0,95*	0,8	0,92	0,94	0,79	0,92	0,94
		RMSE	*448,97*	874,36	547,13	448,28	904,74	542,56	445,46
		MAE	*325,54*	771,21	361,46	324,38	801,82	356,77	317,47
		MAPE	*19,32*	48,05	39,45	17,93	49,4	38,66	16,8
Air-Quality	**Single Layer LSTM**	**R2**	*0,95*	0,83	0,92	0,93	0,85	0,92	0,93
		RMSE	*19,57*	36,36	21,1	23,44	34,26	20,94	23,02
		MAE	*10,58*	31,01	13,23	14,07	27,41	13,02	13,59
		MAPE	*26,98*	92	34,89	47,2	68,11	31,61	40,7
	Double Layer LSTM	**R2**	*0,95*	0,82	0,93	0,94	0,85	0,93	0,93
		RMSE	*19,67*	37	19,65	20,34	33,51	20,2	22,48
		MAE	*10,77*	32,09	12,7	12,47	25,83	12,6	13,26
		MAPE	*33,26*	110,98	41,01	54,18	75,39	38,99	51,44
	Bidirectional LSTM	**R2**	*0,95*	0,83	0,93	0,94	0,86	0,93	0,93
		RMSE	*19,69*	36,38	20,03	21,24	32,61	20,34	22,04
		MAE	*10,79*	31,33	11,81	11,56	24,48	12,28	12,69
		MAPE	*34,46*	108,49	26,97	32,77	72,71	28,81	37,26
	CNN1D	**R2**	*0,94*	0,72	0,92	0,92	0,75	0,93	0,94
		RMSE	*21,56*	46,78	20,89	23,78	43,7	19,25	21,37
		MAE	*12,59*	42,28	13,09	13,98	36,91	11,86	12,41
		MAPE	*46,36*	152,31	37,36	47,17	139,22	33,72	41,1
Solar Generation	**Single Layer LSTM**	**R2**	*0,99*	0,9	0,99	0,98	0,9	0,98	0,98
		RMSE	*12,86*	43,31	11,65	13,78	42,69	12,35	15,46
		MAE	*8,16*	33,14	7,36	8,52	32,54	7,66	9,88
		MAPE	*895897,9*	4382258	7408644	959192,8	4793283	6766378	1530814
	Double Layer LSTM	**R2**	*0,99*	0,93	0,99	0,99	0,93	0,99	0,99
		RMSE	*10,7*	35,21	9,31	10,87	34,5	9,54	11,29
		MAE	*6,12*	27,58	5,42	6,27	28,08	5,61	6,66
		MAPE	*714818,6*	4137325	4734203	536533,8	6136413	4616824	623742,4
	Bidirectional LSTM	**R2**	*0,99*	0,88	0,99	0,99	0,89	0,99	0,98
		RMSE	*13,06*	47,15	10,27	12,43	44,36	11,68	14,8
		MAE	*8,94*	42,34	6,62	7,59	39,5	7,48	9,26
		MAPE	*1550266*	13121066	8379294	1992684	11956808	8446088	1584701
	CNN1D	**R2**	*0,99*	0,92	0,98	0,98	0,92	0,99	0,99
		RMSE	*12,23*	37,83	12,28	14,97	37,46	10,91	13,14
		MAE	*7,11*	28,7	7,91	9,33	28,05	6,71	7,93
		MAPE	*575486,5*	3423600	7936780	1228234	3669001	6307630	1007484