Experimental Neural Networks for Climate Disasters.

Sean Klein, Kevin Russell, Ataish Nehra Northeastern Data Science Capstone
Supervised by: Dr. Fatema Nafa

Abstract: With the rise of natural disasters, it is imperative to anticipate when and how many natural disasters may occur for regions. To this end, we propose a variety of neural network models to better predict future natural disasters based on climate change predictors. Our hypothesis is that as climate change has greater affect on our environment, there will be a growing number of natural disasters that occur, and that by using regression based neural network models, we could accurately predict and represent that trend. The experimental phase for this project can be defined by solving two different problems: the difficulty of handling and reshaping the dataset, and the construction and fitting of three different complex neural networks for accurate prediction. The Neural Networks that were chosen for this experiment are Long Short-Term Memory, Attention, and the Forward-Forward models.

Past related research [1] [2] has focused on predictions for only single countries and specific regions. This project aims to expand the scope of the problem by analysing the weather trends for the top 30 countries as well as the world as a whole.

1 Objectives and Hypotheses.

The objective is to predict Total Disasters with climate change data [3] using LSTM, Attention, and Forward-Forward architectures for time-series and continuous value prediction. Our goal is to evaluate each architecture to determine the good, bad, and ugly among the models for prediction. We hypothesize that the Attention module will be the best model among the three architectures with the Forward-Forward continuous prediction being the worst prediction.

2 Experimental Setup.

This is where we list out our environments and configurations for our experiments within each architecture.

2.1 Hardware & Software

Pre-Processing: Data gathering and pre-processing used the following libraries:
pandas, numpy, scikit-learn, and os

Long Short-Term Memory: Ataish used an Asus ZenbookDuo to build his LSTM model, with 16.0 GB RAM and Intel(R) Core(TM) i7-10510U CPU @ 1.80GHz 2.30 GHz. It also has 64-bit operating system, x64-based processor and a NVIDIA GeForce MX250 graphics card. Jupyter Notebook Version:7.0.6 was used and the following libraries: numpy, pandas, sklearn.compose, sklearn.metrics, sklearn.model_selection, sklearn.preprocessing, tensorflow.keras.layers, tensorflow.keras.models, tensorflow.keras.optimizers, matplotlib.

Attention: Sean performed his model one an alienware pc with NVIDIA GeForce RTX 4060 Ti graphics card, Intel 13th generation Core i7 cpu, and 16 gigabytes of RAM. The code was solely in Python and was created on jupyter notebook in jupyter-lab and stored each iteration on our shared Github. TensorFlow Keras was used for the neural network; pandas, numpy, and sklearn were used for pre and post processing, and matplotlib's pyplot was used for visualizations of the data.

Forward Forward: Kevin trained his continuous prediction with a NVIDIA Geforce RTX 3070 on a AMD Ryzen 7 5800H 3201 Mhz, 8 Cores, 16 Logical Processors. Python 3.9.12 was used with jupyter notebook on jupyterlab to ensure version control with a github repository. The continuous prediction module used torch, typing, torchvision, datetime, traceback, scikit-learn, random, numpy, os, pandas, and matplotlib libraries.

3 Experimentation.

Due to the complexity of the datasets chosen, the first step of the experiment was constructing and pre-processing the dataset so that the models could properly make predictions. Following this, each project member selected a neural network to work on simultaneously; Ataish Nehra worked on a Long Short-Term Memory neural network, Sean Klein worked on an Attention based neural network, and Kevin Russell worked on the Forward-Forward neural network. For comparative results, we used MSE, MAE, and visualizations of the predictions vs. actual.

3.1 Data set.

Description: The dataset was constructed from the following 5 datasets: Surface Temperature, Forest and Carbon, Land Cover Accounts, Sea Level Change and Frequency of Climate Disasters. Originally there was a sixth dataset pertaining to Co Concentrations, however it only recorded the CO concentrations for the entire world, so it did not align with the structure of the dataset that we were constructing. The dataset was created where each row represents the climate records and number of disasters over a year for a specific country. This means that when the dataset is grouped by country then ordered by year, it creates a dataset of multiple small time series of twenty eight years for each country.

Processing: The combined dataset has 899 samples and 30 features where 6 of these features are components of the target variable, Total Disasters, and a 7th feature is year. Mean was used to replace missing values except for Frequency of Natural Disasters where we assumed missing values are zeros. Duplicate and blank rows were dropped as well as same value columns. Luckily no blank rows were found nor was same value columns. IQR was used to detect outliers. All features are in numeric formats so no pre-processing of text or non-numeric values was needed. Data was normalized with MinMax Normalization for time series and Mean Normalization for continuous value prediction. Time series data was grouped into sequences by country, year to train the LSTM and Attention models.

3.2 Models.

Long Short Term Memory: Starting my journey with the ambition to predict natural disasters using climate change indicators, I initiated with a straightforward LSTM model [4]. This model aimed to utilize temporal sequences over a three-year span to forecast disasters. However, I quickly encountered limitations in its predictive accuracy, particularly when trying to generalize across various regions and years. The primary challenges stemmed from the model's simplicity and a limited set of features, leading to significant discrepancies in predictions, especially for unseen data. This experience was eye-opening for me, highlighting the necessity to delve deeper into the data and refine the model's architecture to better capture the underlying patterns.

Moving forward, motivated by the insights gained from my initial attempt, I designed a second model with an enhanced LSTM architecture [5]. This version incorporated additional layers and dropout for regularization, alongside an expanded set of features, aiming for a broader prediction scope over a future five-year horizon. Despite my efforts to improve its sophistication through L1 and L2 regularization, the model struggled with accurate year-specific predictions, evidenced by an increased Mean Squared Error (MSE). This phase was particularly challenging, as it underscored the delicate balance between model complexity and the ability to generalize. The realization that complexity alone wasn't the answer prompted me to rethink my approach and seek a more effective solution.

The breakthrough came with the development of my final model, reflecting the iterative essence of machine learning. By adopting a dual-layer LSTM structure and implementing strategic dropout, but crucially, extending the input sequence to seven years, I was able to capture a deeper understanding of climate dynamics. Notably, I moved away from relying on L1 and L2 regularization, focusing instead on the model's architecture to unearth temporal patterns. This model marked a significant leap in my project, dramatically improving predictive accuracy as evidenced by the reduced error metrics (MAE, MSE, and RMSE). This journey, fraught with challenges and driven by a constant quest for improvement, ultimately led me to a robust model capable of delivering precise, year-by-year forecasts of natural disasters. It underscored the value of persistence and innovation in tackling complex problems, and the final model stands as a testament to the power of iterative refinement in the realm of machine learning.

Attention: The attention neural network [6] was created using TensorFlow and Keras. The model went through many different iterations before its final architecture. The first construction of the model failed in two important ways: mishandling of the dataset and attention layer being improperly integrated into the architecture. I first focused on solving the second issue, I did this by remaking the model where it no longer was sequential, and instead concatenating the base lstm structure with the multihead attention layer. This new model then has a drop out layer and then three different dense models where the output consisted of a single return which is the prediction for how many disasters occurred for that year/country.

The model had a better architecture now but there were still issues with the MSE and Loss metrics exploding in value. To make sure it was not the model, I implemented a basic RNN model and saw that is shared similar results when given the data. This indicated that the pre-processing of the data was improper, so I borrowed the way Ataish was handling his data. Following this, I got much better results with the RNN, so I shifted back to using my Attention model. The output of the Attention model was much better, but had issues with plotting the predictions, so I had to squeeze and reshape the data, which allowed me to create the plot.

With the architecture properly created, I set the epochs hyperparameter to a high number so that I could plot the MSE and Loss score so that I could find the ideal number for epochs which was around 125.

For the attention model, it decided that no feature selection was performed, but instead used the full data set for the analysis. This was chosen under the assumption that having the bigger picture using all the metrics may yield better results. However, in the comparison section of this paper, the reader will show that using a selected sample of the columns received better results.

Forward Forward: The Forward-Forward Algorithm [7] was extended to predict continuous values. Various explorations of the Supervised Forward-Forward Algorithm for classification was made before building the extension to continuous values. The toughest challenge for this extension was breaking away from one-hot encoding data pre-processing and changing the way predictions occurred. Instead of bucketing goodness into a vector to find the biggest goodness, a new method needed to be established to predict continuous values. Ultimately, a new method developed to extend predictions to continuous values, which was tested against the climate change dataset.

The method used three linear layers with hidden layers 256, 128, 128, 1. The data, batch size 32, was trained with 150 iterations at 0.06 learning rate of adam optimizer. Mean Square Error was used as a loss function and the forward-forward thresholds were tuned to 0.07, 4.0, and 0.25 respectively. Data was split 80% train and 20% test dataloaders.

To tune the model, I used the test dataloader to first tune the learning rate by using Mean Squared Error. I later adjusted the forward-forward thresholds to improve the Mean Squared Error by adjusting each layer appropriately. The learning process ran into a lot of local minima, which yielded inconsistent losses, but eventually losses started to increase in value, which suggested over-fitting of the model. Ultimately, the number of epochs was tuned, thereafter, to prevent over-fitting.

Lastly, the model was evaluated with a final test of Mean

Absolute Value, Mean Square Error, and a plot of the actual versus predicted. Each was used to prevent over-fitting.

3.3 Findings

Long Short-Term Memory: The initial model, a basic LSTM configuration, set the groundwork but showed room for improvement with its Mean Squared Error (MSE) of 12.20 and Mean Absolute Error (MAE) of 2.54. Progressing to the second model, we introduced more complexity and regularization in an attempt to enhance performance; however, it resulted in an increased MSE of 34.09 and a slightly higher MAE of 3.81, indicating that added complexity alone did not translate to better accuracy. The final model, however, marked a significant breakthrough, meticulously crafted with an extended input sequence and an optimized architecture, leading to a substantially improved MSE of 7.78 and MAE of 2.13. This evolution not only highlights the importance of appropriate feature selection(i.e. how we used only 12 features including 'Year' and 'Country' as compared to the experiment done for attention model where full dataset was used) and data sequencing but also underscores the effectiveness of our iterative refinement process in reducing error rates and achieving a more accurate and reliable model for predicting natural disasters.

Attention: The Attention model achieved a Mean Squared Error of 13.606 and a Mean Absolute Error of 2.39. This is an improved score in comparison to the simple RNN base model which yielded a Mean Squared error of 15.205 and a Mean Absolute Error of 2.58.

Forward Forward: Results yielded 20.45 Mean Squared Error and 3.09 Mean Absolute Error. A traditional Neural Network was also trained, in comparison, with a Mean Squared Error of 20.26 and Mean Absolute Error of 3.15. Although results aren't that great, future work may include better architectural design or better pre-processing for better performance.

Figures.

Long Short-Term Memory: The below mentioned plots show how our final LSTM model performance is the best as compared to the initial ones.

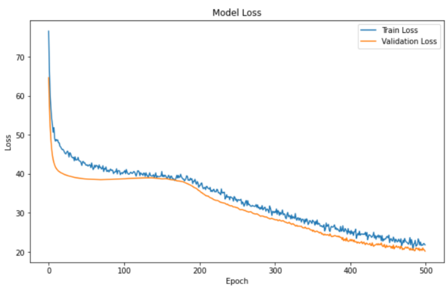

Figure 1: *Training and Validation Loss Over Epochs for Final LSTM Model: This plot traces the decline in both training and validation loss over 500 epochs, indicating the model's learning progression. The convergence of training and validation loss suggests that the model is generalizing well, without over-fitting to the training data.*

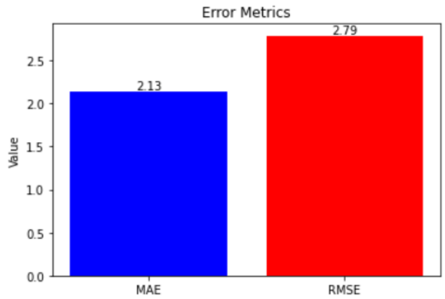

Figure 2: *The chart illustrates the model's Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE), with lower values indicating more accurate predictions. The final LSTM model demonstrates a MAE of 2.13 and an RMSE of 2.79, showcasing its predictive performance.*

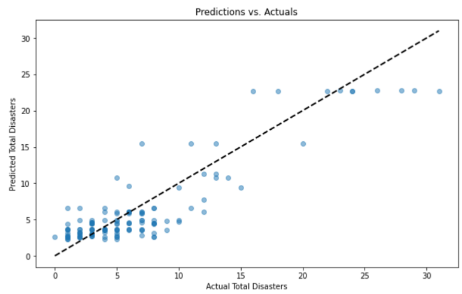

Figure 3: *This graph displays the predicted total number of disasters against the actual recorded events, with the dashed line representing the line of perfect prediction. Points closely aligned with the dashed line indicate higher prediction accuracy of the final LSTM model.*

Attention: The following figures show how the attention model was better able to represent the dataset in comparison to a more simpler model.

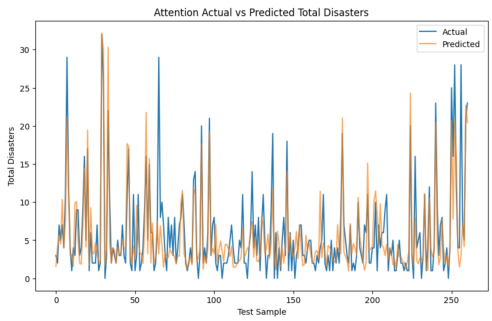

Figure 4: *A comparison of actual (blue) versus attention model's predicted (orange) values pertaining to the total number of disasters that occurred for each country during a four year period.*

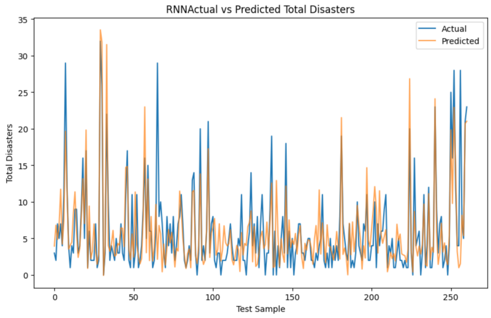

Figure 5: *A comparison of actual (blue) versus RNN model's predicted (orange) values pertaining to the total number of disasters that occurred for each country during a four year period.*

Forward Forward: The below visualizations showcases the predictions on the entire dataset and the distributions of the predictions on a batch of test data.

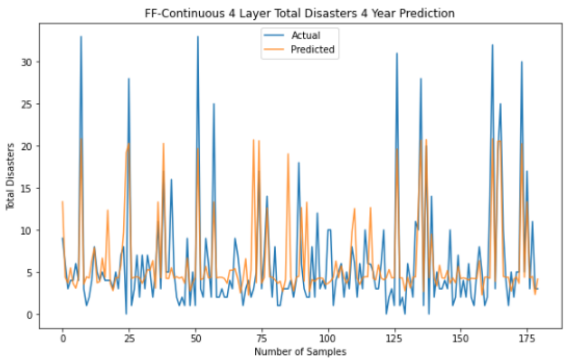

Figure 6: *This is a timeline of actual (blue) versus a 4 year prediction (orange) of Total Disasters with the Forward-Forward Continuous Algorithm using a 4 Linear Layer betwork. A MSE of 20.45 and MAE of 3.15 was achieved*

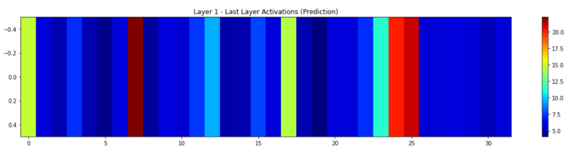

Figure 7: *A showcase of Activations for the last linear layer of predictions using a batch of data (32 samples) shown as a heatmap. Darker (red) colors indicate a higher activation value for the prediction whereas (blue) indicates lower activations.*

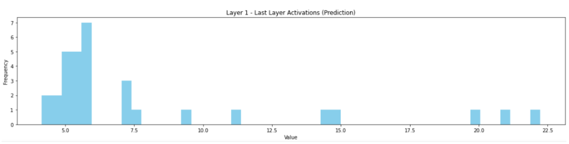

Figure 8: *A showcase of Activations for the last linear layer of predictions using a batch of data (32 samples) shown as a histogram. The histogram shows an overall batch distribution of predictions with the Forward-Forward Continuous method*

Comparisons. The LSTM Neural Network has out performed both the Attention and Forward Forward models. This could be due to the lack of samples of the dataset or it could be due to the feature selection performed for the LSTM model.

The Forward-Forward model had the highest MSE and MAE scores which was expected as it is a relatively new model that's in initial construction and was not intended to solve the time series problem, which was the focus for this project.

The attention model did not take into account the sequential aspect of the project, and the results most likely suffered due to this oversight. But, the model still performed competitively in relation to the other models despite of the sequential problem.

3.4 Conclusion:

In conclusion, all models were able to reflect the different aspects of natural disaster trends. This definitely shows that climate change is having a great impact on the number of natural disasters which occur every year. We conclude that the global community should push towards recording their individual climate change predictors more consistently and accurately so that it can better prepare for the worsening of weather conditions that is happening.

One of the hardest parts of this project was the lack of data surrounding climate change affects; a lot of countries were not tracking such metrics until 1992. This made modeling difficult because we were only able to account for a small year range of 1992-2020. If the global community had taken climate change more seriously earlier, we could have been more prepared for the many disasters that have occurred in recent years.

With the Forward-Forward algorithm, Total Disaster prediction with climate data came out much better than anticipated. The prediction is nearly close to the actual values except for certain areas where the prediction wasn't as good. Neural Activities for the last layer in the architecture shows consistent low values between four to ten with a spike of twenty two for the batch predictions. Obviously, linear layer predictions are not as nearly as good as deep architecture, but does a decent job at predicting Total Disasters.

Next Step: The next step in our project is making more visualizations needed for the final report. Possibly constructing a UI that allows for users to enter in a country and a year, and are able to get back the predicted and actual number of storms. Additionally we may want to explore different kinds of metrics to help us better compare the models that we constructed.

At this point in the life cycle of the project, we are shifting our focus onto the final project write up and presentations so that we can share our findings with our piers and potential researchers in the field of data science and environmental science.

For the attention model, we will use the feature selection performed on Ataish's LSTM model to see if it improves the models performance. This will be done so that the two models can be better compared when the final report is written.

For the Forward-Forward algorithm, to better understand the capabilities of the network and to identify exactly how this network is understanding data in order to improve the architecture.

Future Exploration. If there was more time, we would've liked to extend the number of countries that are represented in the dataset to potentially further increase our accuracy. Increasing the number of countries would also make it so more countries can gain access to accurate forecasting of severe weather that occurs within their borders.

Another exploration would be to make predictions for the specific disasters instead of just the total number. This was not an attainable goal for this current project due to the sparsity of each individual disaster in comparison to the more fruitful TOTAL Disaster column from our dataset. Moreover, future work on predicting the features for the future year and then using those features, we could in turn predict the disasters. This would be a combination of multiple models like supervised learning techniques to predict the features and then using those as input for our LSTM model to predict the natural disasters.

As for the Forward-Forward algorithm, there are a lot to explore, but the most interesting exploration would be to build the current algorithm as an RNN circuit to remember recent and long term memories. Other future exploration would include building deeper neural networks and learning how to properly tune the network.

References

[1] D. Phiri and C. Chembe, "Predicting climate change related extreme natural disasters using machine learning in zambia," *Zambia ICT Journal*, vol. 6, no. 1, pp. 18–24, Dec. 2022.

[2] S.-J. Park and D.-K. Lee, "Prediction of coastal flooding risk under climate change impacts in south korea using machine learning algorithms," *Environmental Research Letters*, vol. 15, no. 9, Sep. 2020.

[3] I. M. Fund. "Climate change data." (), [Online]. Available: `https://climatedata.imf.org/pages/climatechange-data`. accessed: 03.07.2024.

[4] Y. Liu, D. Li, S. Wan, *et al.*, "A long short-term memory-based model for greenhouse climate prediction," *International Journal of Intelligent Systems*, vol. 37, no. 1, pp. 135–151, 2022.

[5] C. Natel de Moura, J. Seibert, and D. H. M. Detzel, "Evaluating the long short-term memory (lstm) network for discharge prediction under changing climate conditions," *Hydrology Research*, vol. 53, no. 5, pp. 657–667, 2022.

[6] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, *Attention is all you need*, 2017. arXiv: `1706.03762 [cs.CL]`.

[7] G. Hinton, *The forward-forward algorithm: Some preliminary investigations*, 2022. arXiv: `2212.13345 [cs.LG]`.