2019 International Conference on Computing, Networking and Communications (ICNC): Green Computing, Networking, and Communications

A Deep Learning-Based Weather Forecast System for Data Volume and Recency Analysis

Jarrett Booz*, Wei Yu*, Guobin Xu†, David Griffith‡, and Nada Golmie‡
*Towson University, MD, USA. Emails: jbooz1@students.towson.edu, wyu@towson.edu
†Frostburg State University, MD, USA. Email: gxu@frostburg.edu
‡National Institute of Standards and Technology, Emails:{david.griffith, nada.golmie}@nist.gov

***Abstract*—Accurate weather forecast is important to our daily life and have both economic and environment impact. Through physical atmospheric models, a short period time weather can be accurately forecasted. To provide weather forecast, machines learning techniques can be used for understanding and analyzing weather patterns. In this paper, we propose a deep learning-based weather forecast system and conduct data volume and recency analysis by utilizing a real-world weather data set as a case study to demonstrate the learning ability of deep learning model. By using the Python Keras library and Pandas library[1], we implement the proposed system. Based on the system, we find out not only the relationship between the prediction accuracy and data volume, but also the relationship between the prediction accuracy and data recency. Through extensive evaluations, our results show that according to the weather data we have been using, more data is beneficial to increasing the accuracy of a trained model. The recency of the data does not have a consistently significant impact on the accuracy of the trained model.**

***Index Terms*—Green Computing, Deep Learning, Big Data, Weather Prediction**

I. Introduction

Weather information is critical to many industrial areas (green computing, smart grid, etc.) and have both economic and environment impact. For example, in the smart grid [1], [2], power generation and consumption prediction depends on the weather forecast information. Thus, weather information affects the optimal schedule of distributed energy resources and storage devices in the smart grid, as well as the reliable energy transmission and distribution [1], [3], [4].

Traditional weather forecast relies on the atmospheric models (e.g., numerical weather prediction model). The data of recent and current state of the atmosphere collected by the weather stations and satellites are the input to the atmospheric models. As atmosphere is a fluid, by solving the equations (e.g., partial differential equations) of fluid dynamics and thermodynamics, the next state of the fluid can be estimated [5]. Nonetheless, these equations are hard to be completely solved and errors will increase with time. In addition, the computation of the atmospheric models requires high volumes of computation resources. Thus, the traditional weather forecast can merely provide accurate weather in a short period times (up to 15 days).

To provide weather forecast in a long period time, machine learning can be used for understanding and analyzing weather patterns [5], [6]. Recent advancements in machine learning have brought neural networks and deep learning systems to the forefront of computer learning technology. Comparing with classic machine learning algorithms, deep learning has proven to be a successful way to analyze increasingly complex data sources and improve accuracy in a variety of applications [7], [8], [9]. Additionally, neural networks can be particularly successful at forecasting in non-linear and time series data sets [10]. Thus, deep learning can be a successful solution to weather forecast [11], [12]. Nonetheless, using deep learning in weather forecast, some questions remain open with respect to the input data: *How many volumes of data shall be input to deep learning system to maximize training accuracy with minimal computational cost? Is older data or recent data more suitable for the training process?* In weather forecast scenario, these questions are crucial to consider when time is a constraining factor or when usable data is scarce.

In this paper, we investigate how to apply deep learning in weather forecast based on collected dataset and find answers to aforementioned questions. Specifically, we develop and implement a deep learning-based weather forecast system with Python Keras library and Pandas library. We use Keras Sequential model as our deep learning model to learn and predict the weather data. The data prediction is based on a sliding window matrix-based mechanism. With the developed system, we discover how the volume and recency of data used for training deep learning model could affect the model performance. We find out how historically recent data performs in comparison to older data. We use the real-world data set from the National Oceanic and Atmospheric Administration (NOAA) National Climatic Data Center (NCDC) for evaluations [13]. The results show that when more data is added in the training process, the accuracy of prediction will be amplified. Additionally, the recency of weather data used in supervised training does not have a great impact on the success of the model.

The remainder of the paper is as follows: In Section II, we provide some background information on deep learning and big data. In Section III, we define the problem and introduce the design of deep learning-based weather forecast system. In Section IV, we show the results of performance evaluation.

[1]Certain commercial equipment, instruments, or materials are identified in this paper in order to specify the experimental procedure adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the materials or equipment identified are necessarily the best available for the purpose.

978-1-5386-9223-3/19/$31.00 ©2019 IEEE

Finally, we provide concluding remarks in Section V.

II. Background

In this section, we introduce the background of deep learning and big data. Deep learning is one technique of machine learning, which is based on artificial neural networks. Generally speaking, neural networks use an array of processors, which are called neurons and connected in a way so that calculations and propagation of results from individual neurons to other neurons can be performed. It is worth noting that the features of interconnecting neurons and the back-propagation of results among them enable the study of architectures, algorithms, etc. [14]. There is a significant amount of research dedicated to using deep learning in state-of-the-art implementations across different domains [15], [16], [17].

Big data can be described as information with a high volume, velocity, and variety of digital information. When it is used together with information technology and analytical methods, some value of the data can be provided [18], [19]. Actually, weather forecast can be regarded as a process of big data analysis. Deep learning environments often require vast amounts of data to be able to perform adequately, making big data a key aspect of deep learning. Nonetheless, one problem with deep learning in a big data environment is to investigate property platforms and algorithms so that the speed of learning process can be improved [20], [21].

III. Deep Learning-Based Weather Forecast System

In this section, we introduce our deep learning-based weather forecast system in detail.

A. Overview

Recall that applying deep learning to help weather forecast, important questions will arise: *How many volumes of data shall be input to deep learning system to maximize training accuracy with minimal computational cost?* Additionally, *how useful is historically old data and is older data or recent data more suitable for the training process? [22]*. To find answers to these questions, we make the following two hypotheses: (i) *Hypothesis I.* We hypothesize that adding more data will increase the accuracy of the predictions made by the deep learning model. This increase may not be linearly correlated with the rate of data increase (i.e. doubling the amount of data may not result in a double of accuracy). Additionally, we presume that there will be a threshold of accuracy that cannot be surpassed, with more data given. (ii) *Hypothesis II.* We hypothesize that more recent data will be more valuable to the deep learning model than older data. More recent data can provide the model with insight into the most current trends, rather than new trends being washed out by historical tendencies that may no longer be relevant.

To test the aforementioned hypotheses, we develop a deep learning-based weather forecast system with several Python libraries: (i) *The Keras library [23]* that is used to implement deep learning architecture for use with the TensorFlow framework; (ii) *The Pandas library [24]* that is used for data manipulation and preparation for input to the neural network.

B. Weather Data

STATION	NAME	DATE	CDSD	EMNT	EMXT	HDSD	TAVG	TMAX	TMIN
USW00093721	BALTIMORE	1940-01	0	5	51	2922	24.2	31.6	16.8
USW00093721	BALTIMORE	1940-02	0	18	56	3796	34.9	42.3	27.4
USW00093721	BALTIMORE	1940-03	0	18	67	4589	39.4	47.7	31.2
USW00093721	BALTIMORE	1940-04	0	28	75	5071	48.9	58	39.8
USW00093721	BALTIMORE	1940-05	57	43	93	5183	63.2	72.7	53.8
USW00093721	BALTIMORE	1940-06	331	50	96	5188	74	83.9	64

Fig. 1. Sample Data

We use real-world data set to evaluate our system performance and test the investigated hypotheses. The data set is from the National Oceanic and Atmospheric Administration (NOAA) National Climatic Data Center (NCDC) [13]. A number of climate data types can be collected via its Climate Data Online (CDO) resource, including local temperature, precipitation, wind, and others. The climate data can be collected in a specific location with a period of time. In this paper, we collected weather data with the features of the Station ID, Station Location, Date, Cooling Degree Days Season to Date (CDSD), Extreme Minimum Temperature for the Period (EMNT), Extreme Maximum Temperature for the Period (EMXT), Average Temperature (TAVG), Maximum Temperature (TMAX), and Minimum Temperature (TMIN) in the Baltimore Washington International Airport (BWI) ground weather station (CDO Label: USW00093721) from January 1940 through December 2017 [13], and data samples are shown Fig. 1.

C. Sliding Window Matrix-Based Mechanism

	Month 1	Month 2	Month 3	Month 4	Month 5
Iteration 1	70	75	78		
Iteration 2	70	75	78	80	
Iteration 3	70	75	78	80	82

Prediction

Window

Fig. 2. Sliding Window Example

We design a sliding window matrix-based mechanism in our system to conduct weather forecast. To predict one month's temperature, the previous 12 months' temperatures are used as input to a deep learning model. Then, the 12 month window could slide along the given dataset, including values that are previously predicted. Fig. 2 illustrates a scaled down version of the sliding window matrix. This figure shows a window size of 2, where the previous 2 months of data are used together to predict the following month. The data in the cells represents the average temperature of the month. The data in the darkened cells illustrate the values within the window that are being used in the current iteration's prediction input. The red data in the cells with darkened boarders shows an example prediction for that cell based on the window being used in the current iteration. Notice that the designed sliding window matrix-based mechanism is used in both deep learning supervised training and prediction phases.

D. Deep Learning Model

We use Keras Sequential model as our deep learning model to build up the neural network. The Keras Sequential model is a part of the Keras framework. Via this framework, stacked layers can be linearly created in neural networks [23]. The model was compiled using the Nadam optimization function. Generally speaking, there are the following three different layers in neural networks: (i) *Input Layer.* The dimensions of the input matrix are used in the input layer. As the window size is set to be 12 in this experiment, the vectors of the input layer are 12. (ii) *Hidden Layers.* The hidden layers in neural networks is densely connected computation layers. Each node in a hidden layer is directly connected to each node in the surrounding hidden layers. Our layer consist of 100 nodes using the Rectified Linear Unit (ReLU) activation function. (iii) *Output Layer.* The result is generated in the output layer. In our case, the single result output is the average temperature prediction.

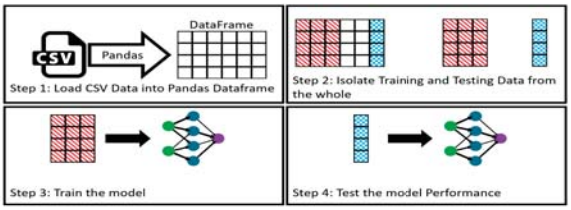

Fig. 3. System workflow

E. System Workflow

As shown in Fig. 3, the system workflow consists of the following four steps: (i) *Step 1. Loading Data:* To build the model and predict values, the CSV file containing the data is read into a Pandas dataframe. The dataframe is then manipulated to isolate data categories, such as the average temperature column of the dataframe. (ii) *Step 2. Separating Data:* The data set is separated into two subsets of data: training data and test data. A given range of years data records are selected from the dataframe and transformed into the sliding window matrix mentioned in Section III-C. Then, this matrix is used as the set of supervised training data for training process. (iii) *Step 3. Model Training:* During the supervised training phase, the deep learning model will make a prediction and make adjustments to its calculations based on the observed error for each iteration. The model's accuracy will increase after each iteration during the supervised training phase due to the continued adjustments. (iv) *Step 4. Model Testing:* After supervised training is complete, test data will be input to the trained model to verify its performance. During this phase, the deep learning model will not make any calculation adjustments based on observed errors.

IV. Performance Evaluation

Via performance evaluation, we measure the effectiveness of our proposed system. The Mean Absolute Error (MAE) is used as a metric to measure the accuracy of the model's predictions, because of the low impact of significant outliers [25]. In our experiment, the size of the window is set to 12, which means that the previous 12 months of data are input to predict the data in the next 1 month. The training data year's range is from 1940 to 2010 and the testing range is set to be the years 2011 through 2017.

Scenario I. To test Hypothesis I defined in Section III-A, the first scenario starts with training a deep learning model with a subset of data that is historically close to the testing data range. In this case, older data is progressively added to the training process. The training sets used in this scenario consist of four different subsets: 10 year, 30 year, 50 year and 70 year subsets of historical data, which are historical data from the years 2000 through 2010, 1980 through 2010, 1960 through 2010, and 1940 through 2010.

Fig. 4 illustrates that the historical data from the years 2000 though 2010 is used to train the deep learning model and the data from the years 2011 through 2017 is used for test the model. From this figure, we can see that the MAE value is 2.95 and the general trend of the data is met. Nonetheless, with the model is hard to predict the peak and valley values (the extreme hot or cold temperatures) accurately, especially temperatures are below 30 °F or approach 80 °F. Notice that we use °F as the unit for temperature to make it consistent with the dataset we choose.

Figs. 5, 6, and 7 demonstrate 30, 50, and 70 years of historical data that is used in the training process, respectively. By using the 30 years training subset, the MAE of the predictor decrease by 0.12, or 4.1 % that is a significant increase in the prediction accuracy of the model. With more data points spanning a longer historical period, the deep learning algorithm is able to calculate a more accurate general trend over time. The MAE of the 50 years test is only 0.02 lower than that of the 30 years test. This demonstrates that adding more data does not always guarantee a significant accuracy increase. The MAE value of 70 years test can achieve 2.74. In this figure, it can be seen that there is a significantly low temperature outlier in the 1950's. This extreme case could have thrown off the model slightly during the training.

From the scenario I results, we can conclude that when more data is used in the model training process, the model predication accuracy increases, which validates that the hypothesis I is true.

Scenario II. To test hypothesis II defined in Section III-A that determines whether a more recent data set is more beneficial to a deep learning model than an older data set of the same size, this second scenario starts with training a deep learning model with a subset of historical data from the earliest years in the total set, and then incrementally increases the training size by adding years of data closer to the testing range. The training sets in this scenario contained four historical data subsets (10 year, 30 year, 50 year, and 70 year data subsets) from the years 1940 through 1950 (Fig. 8), 1940 through 1970 (Fig. 9), 1940 through 1990 (Fig. 10), and 1940 through 2010 (Fig. 7). Because the test of the training subset of years 1940 through 2010 was the same as the final test in Scenario I, the

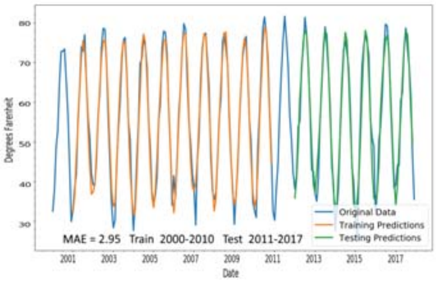

Fig. 4. Training Set of 2000-2010

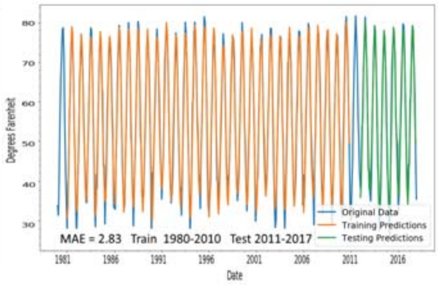

Fig. 5. Training Set of 1980-2010

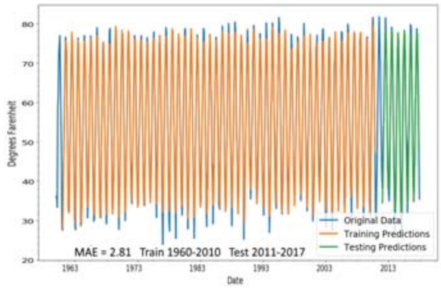

Fig. 6. Training Set of 1960-2010

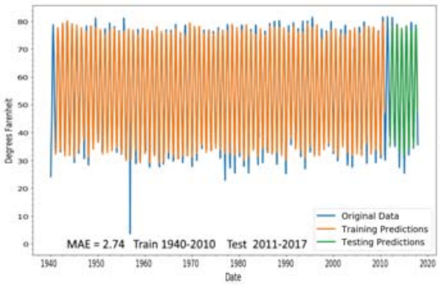

Fig. 7. Training Set of 1940-2010

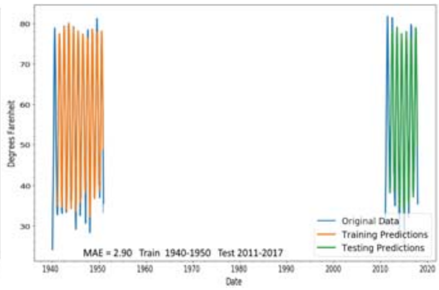

Fig. 8. Training Set of 1940-1950

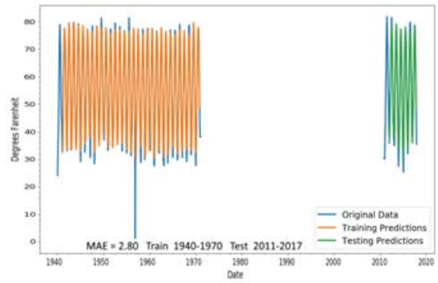

Fig. 9. Training Set of 1940-1970

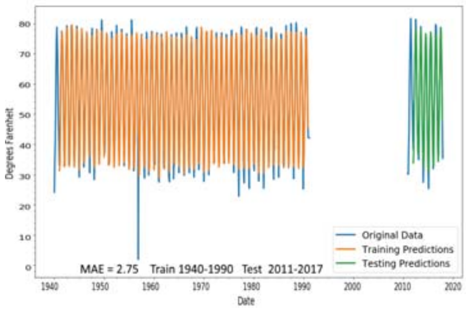

Fig. 10. Training Set of 1940-1990

result is the same as Fig. 7.

In Fig. 8, the MAE of prediction of the average monthly temperature for the years 2011 through 2017 is 2.90. When we train the model with the data from years 1940 through 1970, as shown in Fig. 9, the MAE decrease by 0.1, or approximately 3.45 %. This is a significant decrease in the MAE despite one month in the 1950's having an extreme cold average temperature around 0 °F. Fig. 9 shows the deep learning model has an ability to not be influenced too significantly by extreme outliers, but also demonstrates the model's inability to predict the extreme cases. Also, as shown in Fig. 10, the MAE decrease to 2.75 by adding the additional 20 years of data to the previous subset, which is a decrease of 0.05, or approximately 1.8 % comparing with Fig. 9. This, again, demonstrates that after adding more training data to the model, it can better predict future trends.

Comparing Fig. 8 with Fig. 4, Fig. 9 with Fig. 5, and Fig. 10 with Fig. 6, we can see that the MAE values of the various tests in Scenario II is lower than the corresponding tests in Scenario I. Nonetheless, we cannot say that older data is better at generating a prediction model in all scenarios. Rather, this shows that the older data in this data set, more generally fits the trend of the data seen in the testing set. The more recent data in the training subsets used in Scenario I may have more variance, indicating more extreme temperatures, than the older subsets of data used in Scenario II. This shows that the recency of the data used for training can often be insignificant if the training data meets the general trend of the testing data. As long as the model is able to formulate accurate predictions based on the training data, the recency of the data could be insignificant to the outcome. Thus, the investigated hypothesis II is incorrect.

Overall, these two scenarios show that a large increase in the volume of the average temperature data used for training a prediction model can result in less error when predicting future average temperature values. In general, the recency of the training data, or how close in time the training data is collected in comparison to that of the testing data, may not be greatly significant. The time of data collection appears to matter less than the training data conforming to similar trends that are found in the testing data set. Increasing the volume of data may allow the deep learning model to see wider trends, and therefore make better predictions about future trends.

Results in Different Locations. To further the research and solidify the results, we conduct the deep learning to carry out weather forecast evaluation in different geographical locations: Baltimore in Maryland, Daytona in Florida, Houston in Texas, San Diego in California, and Springfield in Illinois, which represent different areas of the United States with diversified climate patterns. From Figs. 11 and 12, we can see that the four new added cities MAE curve show that the increase

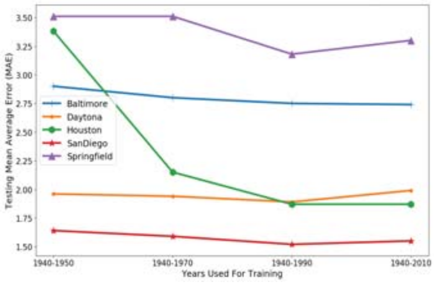

Fig. 11. MAE trend for data from older to newer

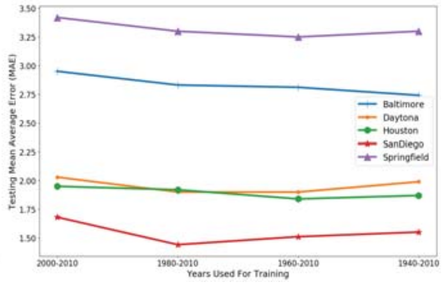

Fig. 12. MAE trend for data from newer to older

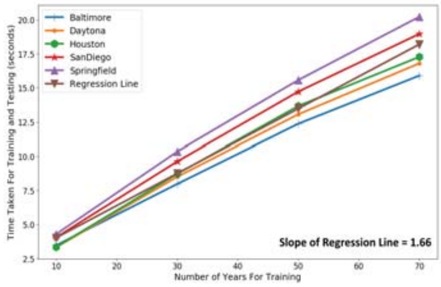

Fig. 13. Average computation time for amount of training data

in training data from 50 years to 70 years actually increase the MAE, which also means decreasing the accuracy of the predictor. This could be due to the fact that the additional data provides some significant data points that skews the model's accuracy. Also, the recency of the data is insignificant in most cases. Fig. 13 illustrates computational cost for training and testing the different sized training sets, which indicates a 166 % increase in computation time when the amount of training data is increased.

V. Final Remarks

In this paper, we developed a deep learning weather forecast system with Python Keras library and Pandas library. Based on this system, we use real-world weather data set to test two proposed hypotheses. Through the extensive evaluations, we found out that increasing the volume of data used to train a deep learning model used for weather predictions could positively impact the model performance. Additionally, the recency of that data used to train the deep learning model is not as significant as the amount of data used. In the future, we will use additional meteorological data sources in addition to the monthly average temperature data used for weather forecast and apply our deep learning forecast system for other big data.

References

[1] G. Xu, W. Yu, D. Griffith, N. Golmie, and P. Moulema, "Toward integrating distributed energy resources and storage devices in smart grid," *IEEE Internet of Things Journal*, vol. 4, no. 1, pp. 192–204, 2017.

[2] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, and W. Zhao, "A survey on Internet of Things: Architecture, enabling technologies, security and privacy, and applications," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1125–1142, 2017.

[3] W. Yu, D. An, D. Griffith, Q. Yang, and G. Xu, "Towards statistical modeling and machine learning based energy usage forecasting in smart grid," *SIGAPP Appl. Comput. Rev.*, vol. 15, no. 1, pp. 6–16, Mar. 2015.

[4] P. Moulema, W. Yu, D. Griffith, and N. Golmie, "On effectiveness of smart grid applications using co-simulation," in *2015 24th International Conference on Computer Communication and Networks (ICCCN)*, Aug 2015, pp. 1–8.

[5] M. Holmstrom, D. Liu, and C. Vo, "Machine learning applied to weather forecasting," http://cs229.stanford.edu/proj2016/report/HolmstromLiuVo-MachineLearningAppliedToWeatherForecasting-report.pdf, 2016.

[6] F. Mateo, J. J. Carrasco, A. Sellami, M. Millán-Giraldo, M. Domínguez, and E. Soria-Olivas, "Machine learning methods to forecast temperature in buildings," *Expert Systems with Applications*, vol. 40, no. 4, pp. 1061–1068, 2013.

[7] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, p. 436, 2015.

[8] J. Booz, J. McGiff, W. G. Hatcher, W. Yu, C. Lu, and J. Nguyen, "Deep learning based mobile malware detection," in *Proc. of IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*. IEEE, 2018.

[9] W. Yu, H. Zhang, L. Ge, and R. Hardy, "On behavior-based detection of malware on android platform," in *2013 IEEE Global Communications Conference (GLOBECOM)*, Dec 2013, pp. 814–819.

[10] X. Qiu, L. Zhang, Y. Ren, P. N. Suganthan, and G. Amaratunga, "Ensemble deep learning for regression and time series forecasting," in *Computational Intelligence in Ensemble Learning (CIEL), 2014 IEEE Symposium on*. IEEE, 2014, pp. 1–6.

[11] M. Prabhat, E. Racah, J. Biard, Y. Liu, M. Mudigonda, K. Kashinath, C. Beckham, T. Maharaj, S. Kahou, C. Pal, T. A. O'Brien, M. F. Wehner, K. Kunkel, and W. D. Collins, "Deep Learning for Extreme Weather Detection," *AGU Fall Meeting Abstracts*, Dec. 2017.

[12] A. Grover, A. Kapoor, and E. Horvitz, "A deep hybrid model for weather forecasting," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2015, pp. 379–386.

[13] "Monthly average data," https://www.ncei.noaa.gov/, Apr 2018.

[14] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural networks*, vol. 61, pp. 85–117, 2015.

[15] W. G. Hatcher and W. Yu, "A survey of deep learning: Platforms, applications and emerging research trends," *IEEE Access*, vol. 6, pp. 24 411–24 432, 2018.

[16] Y. Goldberg, "A primer on neural network models for natural language processing," *Journal of Artificial Intelligence Research*, vol. 57, pp. 345–420, 2016.

[17] J. McGiff, W. G. Hatcher, J. Nguyen, W. Yu, E. Blasch, and C. Lu, "Towards multimodal learning for android malware detection," in *2019 International Conference on Computing, Networking and Communications (ICNC)*, Februry 2019.

[18] F. Liang, W. Yu, D. An, Q. Yang, X. Fu, and W. Zhao, "A survey on big data market: Pricing, trading and protection," *IEEE Access*, vol. 6, pp. 15 132–15 154, 2018.

[19] V. Grover, R. H. Chiang, T.-P. Liang, and D. Zhang, "Creating strategic business value from big data analytics: A research framework," *Journal of Management Information Systems*, vol. 35, no. 2, pp. 388–423, 2018.

[20] X.-W. Chen and X. Lin, "Big data deep learning: challenges and perspectives," *IEEE access*, vol. 2, pp. 514–525, 2014.

[21] W. Yu, F. Liang, X. He, W. G. Hatcher, C. Lu, J. Lin, and X. Yang, "A survey on the edge computing for the Internet of Things," *IEEE Access*, vol. 6, pp. 6900–6919, 2018.

[22] D. Lazer, R. Kennedy, G. King, and A. Vespignani, "The parable of Google Flu: traps in big data analysis," *Science*, vol. 343, no. 6176, pp. 1203–1205, 2014.

[23] "Keras: The python deep learning library," https://keras.io/, accessed: 2018-02-14.

[24] W. McKinney, "Data structures for statistical computing in python," in *Proceedings of the 9th Python in Science Conference*, S. van der Walt and J. Millman, Eds., 2010, pp. 51 – 56.

[25] T. Chai and R. R. Draxler, "Root mean square error (RMSE) or mean absolute error (MAE)?–arguments against avoiding rmse in the literature," *Geoscientific model development*, vol. 7, no. 3, pp. 1247–1250, 2014.