Augmenting Network Performance Datasets with Weather, Sports, and Social Media Data for Improved Predictions

Shivam Dimplekumar Patel

A Thesis
in
The Department
of
Computer Science and Software Engineering

Presented in Partial Fulfillment of the Requirements
For the Degree of
Master's of Computer Science at
Concordia University
Montréal, Québec, Canada

June 2021

CONCORDIA UNIVERSITY
School of Graduate Studies

This is to certify that the thesis prepared

By: **Shivam Dimplekumar Patel**

Entitled: **Augmenting Network Performance Datasets with Weather, Sports, and Social Media Data for Improved Predictions**

and submitted in partial fulfillment of the requirements for the degree of

Master's of Computer Science

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

______________________ Chair
Dr. Tse-Hsun (Peter) Chen

______________________ Examiner
Dr. Yann-Gaël Guéhéneuc

______________________ Examiner
Dr. Tristan Glatard

______________________ Thesis Supervisor
Dr. Tristan Glatard

______________________ Thesis Supervisor
Dr. Brigitte Jaumard

Approved by ______________________
Dr. Leila Kosseim, Graduate Program Director

June 17, 2021 ______________________
Dr. Mourad Debbabi, Dean
Gina Cody School of Engineering and Computer Science

Abstract

Augmenting Network Performance Datasets with Weather, Sports, and Social Media Data for Improved Predictions

Shivam Dimplekumar Patel

Understanding network performance enables network providers to manage their network better. Network performance degradation can lead to network service issues causing monetary loss and customer churn for the network providers. Accurate network performance prediction potentially enables proactive resource allocation to attenuate the anticipated network performance degradation and associated service issues. Previous literature attempted to predict network performance using historical network data. However, real-world network performance is impacted by various external factors. Existing literature fails to consider such external factors that can improve the understanding and predictions of the network performance. This thesis aims to examine if external factors can improve the network performance understanding and predictions. To this end, we inspect the correlation of network performance data with various external data sources such as weather parameters, sports events, and social media posts. Then, we perform network performance data augmentation using the contextual information in such external data. We investigate the network performance prediction improvements using Recurrent Neural Network (RNN) with Long Short Term Memory (LSTM) units after data augmentation. Predictive experiments with data augmentation using NFL sports events highlight a 23% improvement in the network performance predictions. Data augmentation using other external sources considered fails to improve the network performance predictions.

Acknowledgments

First, I would like to express utmost gratitude to my supervisors Drs. Brigitte Jaumard and Tristan Glatard for providing me with the opportunity to pursue a thesis-based program, and thank them for their guidance throughout my Master's degree. I would also like to thank the EXFO team manager, Sylvain Nadeau, for overseeing and supporting my progress throughout this collaborative thesis project. They provided me with important feedback that greatly assisted me during the entirety of the project. Second, I would like to thank my fiance and my family who motivated me to pursue graduate studies in Canada. Everything was made possible by their support, love, and encouragement.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Open Data Opportunities and Challenges

Organizations have been publishing open data on digital platforms to encourage public engagement and research. For example, weather monitoring organizations have published weather records and satellite data to promote open data research. Road accident data and statistics are increasingly published in the digital space for similar reasons. Sports league firms publish game schedules, various statistics and analyses to attract and engage more audiences.

Open data can be independently used to gain results and insights. For example, Krumm et al. [27] provides accurate urban insights using the tweet counts of the areas around New York City. Open data can also be utilized to augment existing data for achieving better results and insights. For example, Yang et al. [49] used financial news from multiple sources to augment historical events data and attempted to improve financial event prediction. Hébert et al. [18] tried to improve the predictability of road accidents for the city of Montreal by augmenting historical road accident data using weather information.

Open data exists in various structures and granularity. Brynjolfsson et al. [13] found a 5-6 percent improvement in output and productivity when the companies adopt data-driven decision-making. Data augmentation using irrelevant data can degrade the results, incurring resource wastage. Hence, it is crucial to identify the potential use of open data for every use case before using them to gain insights.

1.2 Network Performance Prediction

Network service providers sell their services with specific requirement thresholds called Service Level Agreement (SLA). A failure to provide adequate network service can yield customer churn and monetary loss for service providers. Service providers want to retain and attract customers to minimize such losses.
Previous Research has attempted to address this issue by predicting the most likely churners (customers) using the customer interaction data [9, 10, 36]. Another area of research has aimed to improve network traffic prediction using historical network traffic data to improve traffic management and Quality of Service (QoS).

Network performance can be quantified by various network Key Performance Indicators (KPI). Network delay KPI indicates the time required by a packet to reach a destination from the source. This KPI is called "Latency" in the literature. Packet loss KPI is the difference between the packets sent by the source and the packets received by the receiver. An increase in these KPI values indicate poor network performance resulting in inadequate service. Estimating the network performance can enable the service providers to manage network resources better and proactively address performance issues.

Past research has also attempted to estimate the network performance by predicting one or more network KPIs using historical network KPIs data [24, 39, 41, 35]. Network performance is directly impacted by the number of active users and their usage (Web Browsing, Video Streaming, etc.). Since most of these users are human beings, factors influencing their network needs can significantly impact network performance.

External events and natural phenomena can directly affect people's activity and their network usage. For example, a concert attracting thousands of people might increase the local network usage, causing higher service delays and poor network performance.

Consequently, understanding the network performance and such impacting factors has multi-fold benefits. Improved performance understanding can be translated to improved future network performance predictions. Service providers can enhance their network resource management and proactively allocate resources to attenuate the predicted network impact.

Existing studies predicting network KPIs leverage temporal patterns in historical network data. Such data might lack information regarding external factors responsible for changes in the network performance. Consequently, these studies did not leverage the knowledge of such external factors to improve predictions. Little research has aimed to identify, understand and link the network performance behaviour, to external real-world factors, for improved predictions.

1.3 Problem Statement

In this thesis, we examine if certain external factors can improve the network performance understanding and predictability in terms of network packet loss.

To this end, we identify certain external factors that potentially impact network performance. We perform a correlation analysis between the external factors and the network KPIs to assess the strength of similarity between the changes in network KPIs and the external factors within a specific area. We augment packet loss data using external data to improve packet loss predictions. We compare the predictions of Deep learning models trained using the augmented data vs. the predictions of models trained with only packet loss data.

1.4 Thesis Contributions and Outline

The contributions of this thesis are the following:

- We analyze the correlation between weather parameters and KPIs;
- We show the benefit of augmenting network packet loss with information on sports events for better predictions using Deep learning;
- We present a set of experiments augmenting network packet loss with social media (Twitter) data.

This thesis is organized in six chapters. Chapter 2 provides an overview of the existing literature in network KPI prediction. It also discusses the predictive analysis enabled using external data like weather, sports events, and social media. Chapter 3 details the experiments to correlate the network performance behaviour with the weather parameters. Chapter 4 identifies the sports events with potential impact on the network performances and presents the events data. Then we discuss the performed statistical analysis to assess the impact on network performance during identified sports events. Then we describe the network data augmentation using the sports events data and present a predictive experiment using LSTM models to quantify the network "packet loss" prediction improvements because of sports events data augmentation. Chapter 5 uses the Twitter data and assesses the strength of correlation with the network variables by correlation analysis. Then we augment the network data and describe a predictive experiment using LSTM models to quantify the network "packet loss" prediction improvements because of Twitter data augmentation.

Chapter 2

Related Work

This section reviews the studies covering network KPI prediction and predictive analytics enabled by weather, sports events, and social media data.

2.1 Network KPI prediction

Raca et al. [39] utilized Random Forests, AdaBoost, Gradient Boosting, Support Vector Machine, Gaussian Process, Neural Networks and other conventional algorithms like Autoregressive Integrated Moving Average (ARIMA) to compare the performance for cellular throughput prediction using user device based metrics. They conducted a simulation using ns-3 framework to collect the simulated data. However, a summary of utlilized datasets could not be derived from this work. Random Forest performed the best, and the ARIMA model performed slightly worse than the Random Forests at different sampling frequencies. The study also noted that SVM and Gaussian Process generated the highest prediction Mean Average Error. The study concluded that integrating measurements from the network with measurements from the end device would not introduce any significant benefits due to the high predictability of static devices throughput but helped reduce the throughput prediction error by 25 % for mobile devices (using Random Forests).

Sayeed et al. [41] used the ARIMA model to predict the wireless link quality metric at the application level. The metric used was the signal-to-noise ratio. The study use an Alcatel-Lucent sample level simulator to generate device metrics with a frequency of 20 millisecond (ms) for 2400 seconds. The study observed that link quality prediction accuracy varied between 90 to 97 % depending on the prediction horizons.

Pierucci et al. [37] used Learning Vector Quantization Neural Network to provide a classification of warnings related to problems due to high traffic and bad quality of radio channel. The authors used Key Performance Indicators related to active users and Channel Quality Indicator, derived from the counters located on network equipment of Italia Telecom. The training data spanned 14 minutes with a 4 ms frequency. The test results for an entire day showcased only 0.7 % false positives in detecting the warnings. The presented neural network enabled better real-time network monitoring to automatically detect the congestion to improve the user-perceived quality of service.

Bhorkar et al. [11] proposed a hierarchical Deep learning framework using LSTM for real-time prediction of cell load and radio channel quality KPI using historical KPI data. They collect the network measurement data from cellular networks and Cell Traffic Recordings data from various network management systems. They collected three months of data within same geographical area containing about 1 million records with a frequency of 15 minutes. The proposed framework embeds the weather, day of the week, and holidays information using a separate neural network. The framework successfully reduced the Root Mean Squared Error by 15 % for real time short term load prediction compared to Random Forest and XGBoost methods. This methodology also reduced the real-time long term cell load prediction RMSE by 32%.

We observe that different network KPIs are utilized to measure and predict network performance and network Quality of Service (QoS). Due to the easier availability of computational resources, studies are increasingly using Deep Learning techniques to obtain better results. Many studies base their results on simulated network performance data due to a lack of open data availability. The simulation data might not accurately capture the real-world characteristics limiting the use of such research. Many reviewed studies fail to adequately summarize and quantify their results, hindering replicating and extending research using existing literature.

2.2 Predictive analytics enabled by Weather, Sports, and Twitter data

Koesdwiady et al. [26] attempted to improve the traffic flow prediction using localized weather information. The weather data from 16 weather sites are hourly sampled and the traffic data is sampled every 15 minutes at 47 roads. The weather data is linearly interpolated to achieve 15 minutes sampling frequency . The authors performed a correlation study between traffic and weather variables. Both the data are collected in the weekdays and weekends from August 1, 2013 to November 25, 2013. They proposed a deep belief networks architecture to enhance the traffic flow prediction accuracy using the historical traffic flow and current weather data. The proposed architecture outperforms the ARIMA (0.26) and Artificial Neural Networks (0.08) models by achieving a 0.07 Root Mean Squared Error error.

Klein et al. [25] presented a predictive model built using the Weather Impacted Traffic Index toolset to predict the airport delay using weather forecast data. While the study did not specify the exact data time length used for their experiments, they presented the visual plots for the weather and airport delay for the year 2007. This study analyzed the weather impacts by twelve types of weather (snow, storm, wind, etc.) to improve the delay predictions. The presented approach could predict the delay with 80 % accuracy.

He et al. [17] used the road traffic-related content on Twitter to improve the traffic predictions. They collected the hourly traffic measurements from 943 loop detectors between August 3, 2011 and September 30, 2011 using California Performance Measurement System. The tweet data is also extracted for the same time and pre-processed to remove unwanted elements. They performed a correlation analysis to identify the potential similarity between four weeks of traffic measurements and tweet counts based in the San Francisco Bay area. Then the study provided an optimization framework that extracts the tweet semantic-based traffic indicators for each time-point. They built an auto-regression model using only traffic data (base) and another model using two different types of semantic-based traffic indicators (enhanced) to predict the traffic. The enhanced model produced 4.51 Mean Absolute Percentage Error (MAPE) while the base model produced 5.18 MAPE.

Ranjan et al. [40] presented a sentiment-based prediction model to estimate the subscriber growth rates for five telecom companies in India. The study collected tweets related to five Indian telecom operators from March to July, 2017 and performed Ontology assignment and semantic analysis of the tweets to categorize them into five types of opinion metrics. Such categorical opinions were used to predict a sentiment score and growth rate associated with each telecom company. This approach predicted the growth rates for each of the four months with 90 % correlation strength with the actual growth rates.

Wang et al. [46] successfully utilized the Twitter posts data to improve the crime predictions using Latent Dirichlet allocation and a generalized linear regression model. The twitter activity of a news media agency was collected for the period of February 22, 2011 till October 21, 2011. The ground-truth crime data were obtained from local law enforcement agencies, focusing on hit-and-run incidents during the above mentioned time frame. The authors analyzed and combined the semantics of the tweets with the historical hit-and-run incidents data to attain a better predictions. The proposed approach successfully identified 25 % additional incidents than the baseline model.

Sport events data has been used extensively to model their impact on tourism, national and local economy [29, 16]. Such studies do not aim to augment existing data to improve the results. However, various sporting events account for increased physical traffic, online activity, and interactions. Hence, there is merit in evaluating the potential impact on network performance during such events.

While abundant studies utilize various open data to augment a primary dataset and improve results, we observe a shortage of external data-aided research to understand the external factors impacting the network performance and use such insights to improve performance prediction.

Chapter 3

Weather data for improved Network Performance predictability

Weather affects the decisions of people in a variety of instances. For example, people avoid going out on rainy days and spend time at their homes streaming video content of different online media. A similar observation holds for scorching and humid days when people are advised to be indoors. An upcoming storm encourages employees to work from home and attend meetings over the network. The survey in [3] notes that as of 2019, more than 85 percent of adults in the United States (18 or older) have an internet subscription. The 2020 study in [43] reports that an average household in the US uses 3.5 times the amount of data they used five years ago and 38 times the amount they used ten years ago. An average network user consumes more bandwidth to access Netflix, YouTube, and other video streaming applications. Hence, an additional amount of time spent by people at home due to unpleasant weather conditions results in higher (than usual) network and data usage. We hypothesize that such a change in network usage due to weather will impact the overall network performance. This chapter investigates if weather data can be used to improve prediction of the network performance.

We first determine the weather and network variables to be used for the analysis. Then a statistical analysis is performed using the selected variables to detect a correlation between weather and network variables. Based on the correlation results, we evaluate the issue of improving network performance predictability using weather variables.

The rest of the chapter is organized as follows: Section 3.1 describes the data and preprocessing done for this study. Section 3.2 describes the visualizations and seasonal adjustments performed for the analysis. Section 3.3 discusses correlation aspects. Section 3.4 provides the correlation results and observations used to determine the usability of the weather records to improve network performance estimation. Section 3.5 presents inferences and conclusions obtained from the correlation results.

3.1 Datasets and Preprocessing

This section describes the data and preprocessing steps performed for this study. Then, it covers the network monitoring data with its preprocessing and later presents the weather data with the appropriate preprocessing steps.

3.1.1 Communications Service Provider (CSP) monitoring

The CSP monitoring data is collected through Ethernet Service Operations, Administration and Management (SOAM) active test [22]. The SOAM test is a standardized test method used to provide health assessment of Ethernet services. One of the functionalities of this test is to record and check network performance to compare with the Service Level Agreement requirements. The SOAM test is usually conducted at the beginning of the network deployment and at regular intervals after deployment to monitor the fulfilment of the SLA requirements and detect service faults.

CSP Data

The data spans over six network sites. One of the goals of this test is to ensure network performance for all the Service Level Agreements. Each Service Level Agreement is tested using one or more tests. Multiple tests for different Service Level Agreement are conducted at a time interval of one minute where a unique test id identifies each test.

The data records start in December 2017 and are collected until May 2018. The dataset contains following variables:

1. Time stamp: Time of test execution.
2. clli site: Location of sender node.
3. Service Level Agreement (SLA): Id of the Service Level Agreement targeted by the conducted tests. The dataset contains tests for 38 SLAs.
4. Test id: Id of the test conducted. Multiple tests are conducted to check the fulfilment of the requirements for each SLA. There are approximately 300 test ids for each SLA.
5. Delay: Round trip delay. It is the time taken by a packet to travel from the sender node to the destination endpoint and back to the sender. There are three fields associated with delay: average, minimum, and maximum. This field is also referred to as "Latency."
6. Variational Delay: Variation in the delay values across multiple packet transmissions. There are three fields associated to Variational Delay: average, minimum, and maximum.
7. Transmitted packets: Number of transmitted packets by a sender node.
8. Received packets : Number of received packets by the sender node (after completing one round trip).

9. Packet loss: Number of lost packets during a round trip test.

Preprocessing

We group the data using the network site associated with each data point to create a separate dataset for all six network sites. The per-minute sampled data is aggregated to hourly samples by summation at each network site. This procedure aligns the frequency of network (SOAM test) data to the weather data.

With the help of industry experts, we identify three parameters out of the above-described parameters that are more likely to reflect the network state (network performance) at each network site. We perform a correlation analysis among these three parameters to identify redundancy. Table 1 shows the correlation matrix of the selected parameters at one of the network sites.

Table 1: Network Parameters correlation matrix at a network site

	packet_loss	delay_avg	delay_var_avg
packet_loss	1.0	0.51	0.73
delay_avg	0.51	1.0	0.71
delay_var_avg	0.73	0.71	1.0

We observe that the "delay_var_avg" has a high correlation with "packet_loss" and "delay_avg". Similar correlation results are observed with the dataset associated with the rest of the five network sites. Hence, we retain "packet_loss" & "delay_avg" and eliminate "delay_var_avg" from further analysis to avoid the generation of redundant results.

The plots in Figure 1 show the scaled (min-max normalization) hourly packet loss and hourly average delay recorded at one of the network sites. Both the variables exhibit high seasonality with occasional peaks in packet loss and troughs in the average delay time series. We exclude certain time points with high value to highlight the patterns of both variables.

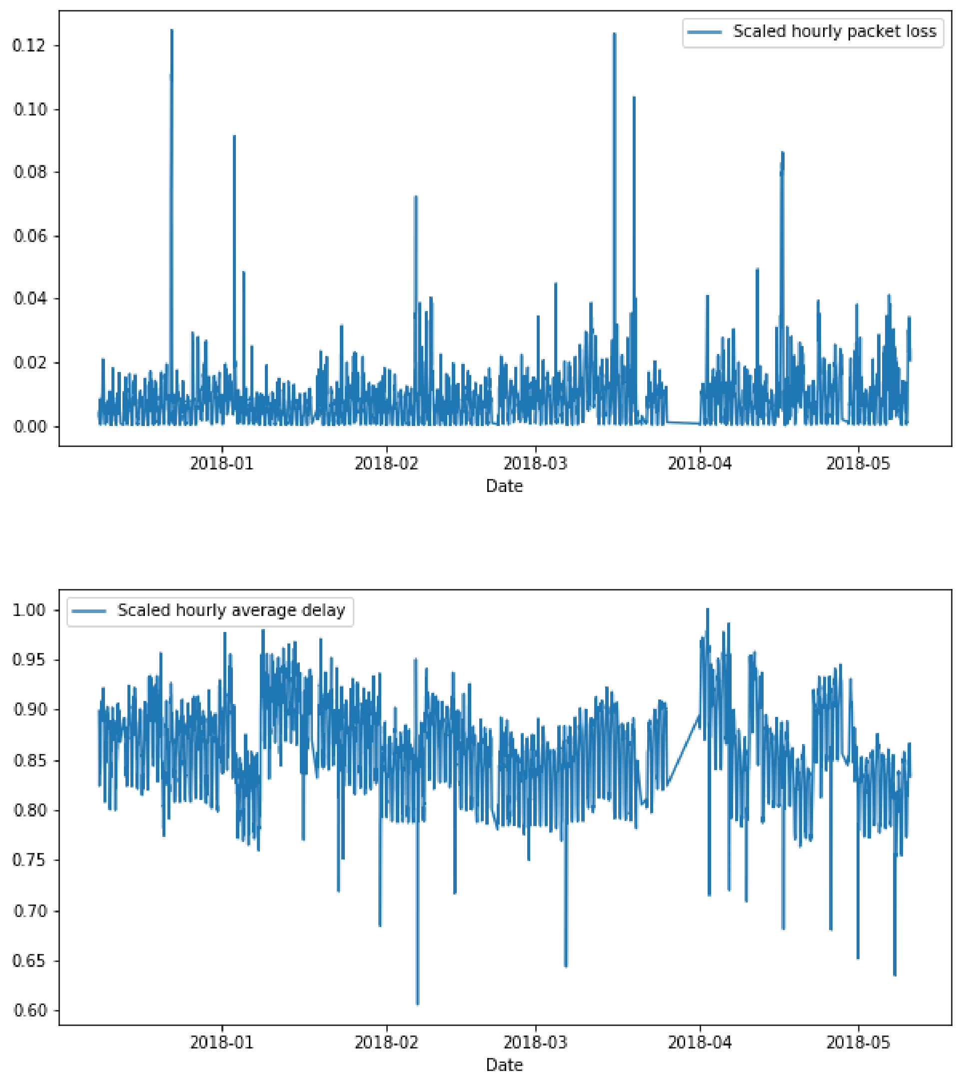

Figure 1: Scaled Network Performance

3.1.2 Weather

We use weather data from [2]. This website offers hourly weather data for multiple cities in the USA. We select cities closest to the network site locations and collect the weather data for each selected city. We only collect the weather data from December 2017 to May 2018 to align the data with network data.

Data

The Weather data contains the following variables:

1. Site: A four-letter representation of the city of site location
2. Date: Date of the observation
3. Hour: Hour of the observation
4. Temperature: Observed temperature at given time point
5. Dewpoint: Temperature at which the water vapor would become liquid
6. RH: Relative Humidity
7. WindDir: Wind direction
8. Windspeed: Wind Speed
9. CldFrac: Fraction of sky covered by cloud
10. MSLP: Mean surface level pressure
11. Weather: A textual representation of the type of observed overall weather
12. Precip: Amount of Precipitation in the atmosphere
13. Source: Either recorded by instruments or filled using interpolation

Preprocessing

Daily temperatures and wind are the most fundamental phenomena that affect peoples' decision-making. Heyes and Saberian [20] studied decision making of 207,000 court cases and concluded that an average increase of 1 Fahrenheit temperature resulted in the reduction of positive outcomes by 8.5 percent. Cheema and Patrick [15] demonstrated that warm (vs.

cold) temperatures deplete human resources, influencing peoples' performance on complex choice tasks. The authors discover that Tokyo people were less likely to keep travelling on calm (<2km/h wind-speed) days. They also observed that severe wind and weather conditions interrupt the routine behaviour of people. Based on such pieces of evidence, we select the temperature and wind-speed parameters for our analysis. To eliminate redundant weather features, we inspect the correlation matrix generated using the weather data. The features strongly correlated to the temperature and wind speed are excluded from further analysis.

Table 3 presents the weather correlation matrix at one weather site. We observe that Dewpoint is highly correlated with temperature. Hence we do not consider Dewpoint for the rest of the analysis. We also limit the number of features for this study by eliminating RH and MSLP features. Further analyses are conducted using temperature, wind speed, CldFrac, and Precip weather features.

Table 2: Correlation matrix for a weather site

	Temperature	Windspeed	RH	Dewpoint	CldFrac	MSLP	Precip
Temperature	1.0	0.26	-0.003	0.81	0.09	-0.16	-0.15
Windspeed	0.26	1.0	-0.16	0.16	0.16	-0.03	0.015

3.2 Visualization and Seasonal Adjustment

Figure 2 presents a line plot of hourly temperature and hourly packet loss at one of the six network sites. We observe that values of both variables tend to increase as the day progresses and starts to decrease in the latter half of the day. This constitutes a seasonality because the variable values follow a cyclic pattern every day. Natural phenomena predominantly govern such seasonality in weather variables. In contrast, the cause of seasonality in network performance data is quite intuitive as most people sleep throughout the night, reducing the network load and increasing the network usage during the daytime.

Such a seasonality might lead to a strong correlation among the variables that dominate any hypothetical similarity introduced by any other causal relation. To determine such latent similarity, we remove the seasonality from the relevant variables.

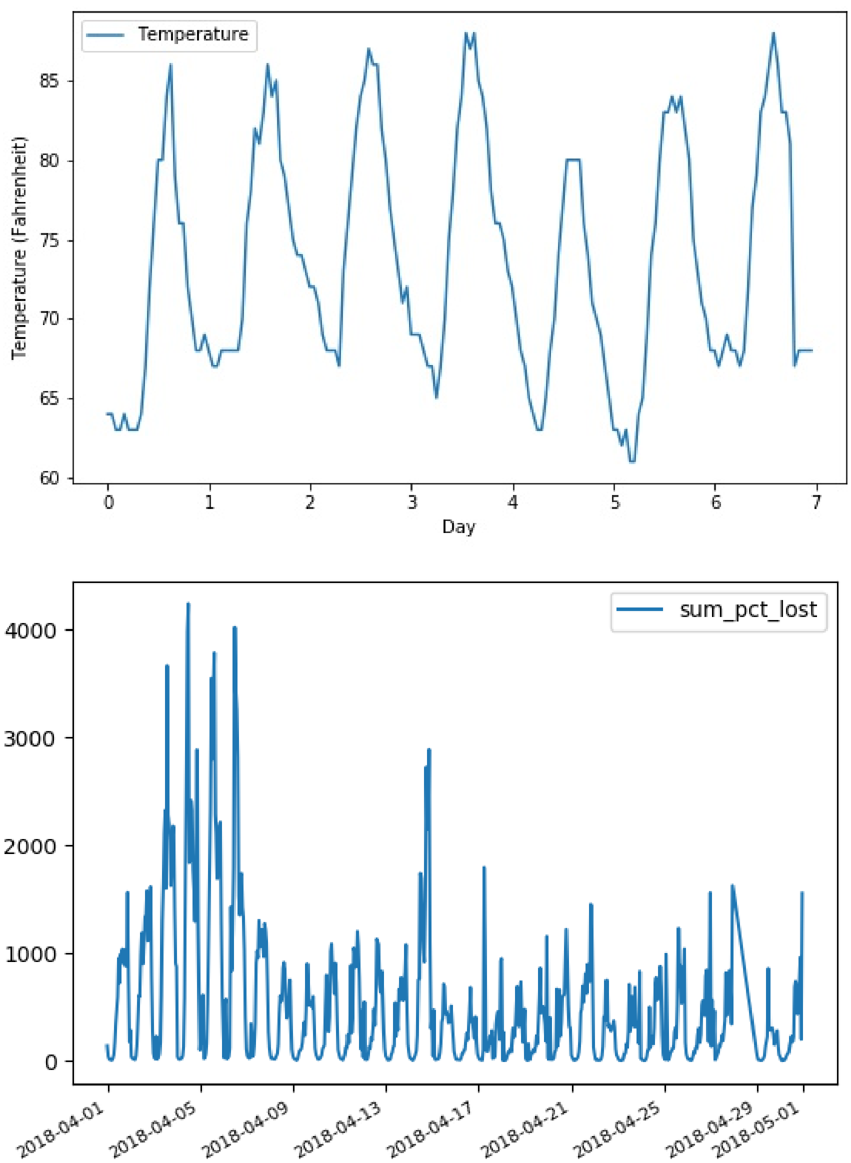

Figure 2: Hourly Temperature and Packet loss

3.2.1 Seasonal Decomposition

Seasonal Decomposition is based on the notion that any time series can be separated into following three components.

1. Trend Component

   It is a long-term increase or decrease in the data values. It can be imagined as a long-term path of a time series. The trend value for each time-point is calculated by averaging "m" values centred at the time-series value of each time-point. The length of this moving average window can be customized to get the trend component of various granularity.

2. Seasonal Component

   The patterns that repeat themselves at a constant frequency/time are seasonal components. A time series may have more than one seasonal component with different periods of seasonality. The trend component is removed from the time series to get a "Detrended series." Each time-point seasonal value is calculated by averaging all the values with the same time-points in the detrended series.

3. Residual Component

   The component that does not contribute to the trend or the seasonality is the residual component of the time series. This component is calculated by removing the trend and the seasonal components from the original time series.

The Statsmodel library [6] is used to implement seasonal decomposition. The library provides two methods for the task at hand.

Seasonal_decompose: This method is a naive implementation of decomposition using simple moving averages for the given periodicity.

STL: It uses Locally Estimated Scatterplot Smoothing (LOESS) to extract smooths estimates of the three components. This method consists of number of parameters to fine-tune the decomposition and get optimal results.

Both methods provided equivalent results in terms of end correlation results. Hence, results with only "Seasonal_decompose" method is used for further analysis. Figure 3 shows the decomposed temperature and packet loss variable using seasonal_decompose method.

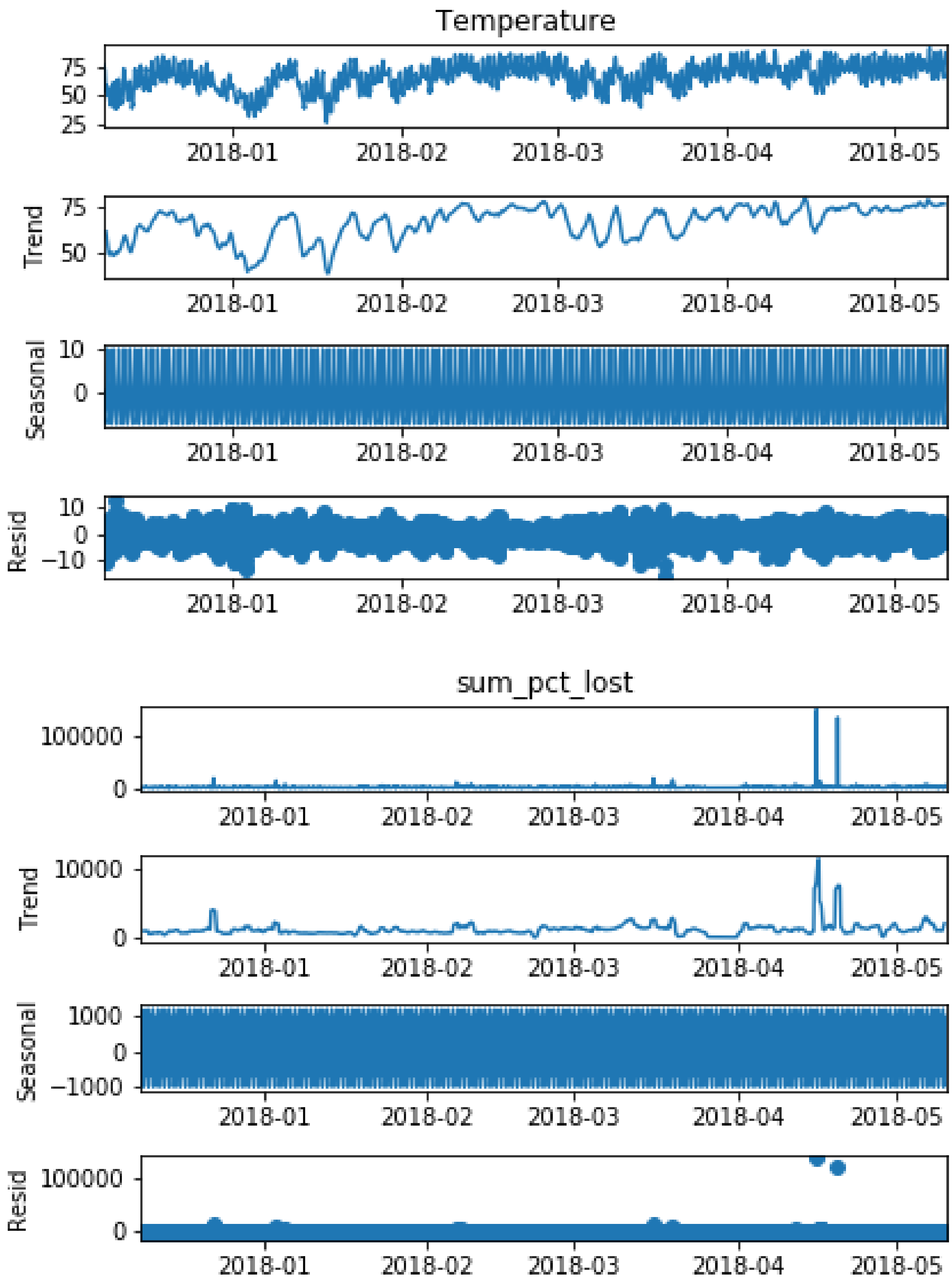

Figure 3: Seasonally decomposed temperature and packet loss

3.3 Correlation

The association between two signals can be measured with the correlation coefficient. The value of such coefficient lies in the range [-1,1]. +/-1 indicates a perfect correlation between two variables. The strength of such association decreases as the value shifts towards 0. There are three extensively used statistical correlation coefficients, explained briefly in the following sections.

3.3.1 Pearson Correlation (Pearson r)

The Pearson correlation measures the degree of the association between linearly related variables. This metric is sensitive to outliers and only measures linear correlation. For two discrete time series x and y of length n, the Pearson correlation is given by the equation:

$$\rho_{xy} = \frac{\sum_{i=0}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=0}^{n} (x_i - \bar{x})^2 \sum_{i=0}^{n} (y_i - \bar{y})^2}}$$

Here the terms $\bar{x}$ and $\bar{y}$ are the mean values.

3.3.2 Spearman Rank Correlation

Spearman rank correlation measures the monotonic relationship between two variables. The Spearman correlation between two variables is equal to the Pearson correlation between the rank values of those variables. The rank of a point is its position when the data is sorted in ascending or descending order.

This coefficient is not highly susceptible to outliers because the ranks are lower-bounded by zero and upper-bounded by the data length.

3.3.3 Kendall Tau Correlation

Kendall Tau correlation also assesses the similarity of two variables based on the rank of the data points. However, it uses a different approach than Spearman correlation to calculate the coefficient. This coefficient is given by the following equation:

$$\tau = \frac{(\text{ number of concordant pairs }) - (\text{ number of discordant pairs })}{\begin{pmatrix} n \\ 2 \end{pmatrix}} \tag{1}$$

For two time-series x = $x_0, \ldots, x_n$ and y = $y_0, \ldots, y_n$, a data-point pair (x_i, y_i), (x_j, y_j) is concordant if:

$$(\ x_j > x_i \text{ and } y_j > y_i\ ) \text{ OR } (\ x_j < x_i \text{ and } y_j < y_i\ ) \tag{2}$$

The pair is discordant if:

$$(\ x_j > x_i \text{ and } y_j < y_i\ ) \text{ OR } (x_j < x_i \text{ and } y_j > y_i\ ) \tag{3}$$

And the pair is tied if:

$$x_{t+i} = x_t, \text{ or } y_{t+i} = y_t \tag{4}$$

The denominator represents all the possible combination of pairs (x_i, y_i), (x_j, y_j). This correlation is not susceptible to outliers since the rank of data points is used to determine the strength of similarity.

The correlation between network data and weather parameters is likely to be non-linear, and to be affected by outliers. Puth et al. [38] examined the performance of rank correlation coefficients to describe association strength. They observe that any tie in the ranks of data decreases the confidence of the Kendall Tau coefficient compared to the Spearman coefficient. Hence, we use Spearman Rank correlation to generate a robust similarity metric for our analysis.

3.4 Numerical Results

We decompose all sites' network and weather data into the seasonal, trend, and residual components using the seasonal decompose method described in the Section 3.2. The seasonal component represents the daily cycle which is not pertinent to this study.

We perform a correlation analysis between the trends of network parameters and trends of weather parameters. Table 3 presents these correlation results.

Table 3: Spearman Correlation results for data from December - May

Trends	pct_lost				delay_avg			
	Temp	Wind	Precip	Cloud	Temp	Wind	Precip	Cloud
Site1	0.16	-0.11	0.06	-0.05	-0.12	-0.14	0.12	0.3
Site2	-0.13	0.02	0.08	0.08	-0.32	-0.08	0.12	0.14
Site3	0.39	-0.02	0.0	-0.05	-0.2	-0.02	0.15	0.07
Site4	0.06	-0.08	-0.08	-0.11	-0.21	-0.25	-0.03	0.04
Site5	0.17	-0.04	0.11	0.07	0.22	-0.01	-0.01	0.0
Site6	0.21	-0.1	0.05	-0.23	0.1	-0.17	-0.01	-0.06

We try to detect any impact on network performance when the network users adapt to weather changes by observing latent similarities in weather and network performance data.

From the results in Table 3, we observe a consistently negative correlation of network parameters and wind speed which suggests that an increase (resp. decrease) in wind speed happens simultaneously as an decrease (resp. increase) in packet loss and average delay values.

Intuitively, Network users would prefer to stay indoors during unfavourable high-wind conditions, increasing the network load, and choose to go out when the weather is calmer, reducing the network load. The observed correlation indicates the opposite. However, the magnitude of the observed correlations is very low for conclusive arguments and results.

The correlation of temperature vs. packet loss suggested an increase in temperature and increased packet loss simultaneously at all network sites except “Site 2,” where the correlation is negative. It might indicate that network usage increases during very hot periods as network users choose to stay indoors, potentially increasing packet loss.

The correlation of temperature vs. average delay suggests an increase in temperature and decreased average delay co-occur at all network sites except “Site 5” and ”Site 6”. This might mean that the average delay decreases because less network usage is observed as more people choose to perform activities excluding network usage.

These observations are contradictory because network performance variables measure network degradation and ideally increase/decrease simultaneously.
To summarize the results in Table 3, all the correlation values between trends of weather and network variables are very low to derive fruitful conclusions.

3.5 Conclusion

Experiments did not allow the detection of a significant similarity between trend components of network performance and weather features. Owing to the above discussions, we conclude that the considered weather parameters do not have potential value in improving our dataset’s predictability of network packet loss.

Clustering the correlation analysis of sites with similar geographical features can help explain the variability of correlation results. Our study measured monotonic relationships between different data. Non-linear analysis considering various natural factors spanning multiple years that potentially affect network user behaviour may extend the findings of this study. However, for the scope of this study, we move on to investigate other variables that potentially have a significant impact on network performance.

Chapter 4

Sports events for improved Network Performance predictability

Sports events have effects ranging from short-term employment generation to economic and social development of neighborhoods to attract the event's organizers. Charlebois and Stevens [14] examined the impacts of sports events in the Niagara region of Canada and provided qualitative social and economic impact by surveying 251 people. They determined that the considered events had a positive perceived social and economic impact.

Athletes, participants, officials, and fans travelling to attend sports events constitute sports tourism. The report in [5] found that travellers attending sports events increased by more than 10 million since 2015 to a total of 169 Million. The study reported a business sales impact of 3.5 Billion USD and a labor income impact of 0.9 Billion USD in the communications sector due to sports tourism.

The website in [1] reports the top 50 sport events viewerships in the USA for 2018. The viewership includes television broadcasts and online streams. Upward of 20 Million views are generated for almost all 50 events.

Since sport events generate a lot of physical and digital traffic, we hypothesize that a sporting event will significantly impact people's behaviour, affecting telecommunication network usage and performance.

This study identifies the type of sport events with potential impact on network performance and conducts statistical tests to quantify the network performance impact during the event time. Then we utilize a Deep learning technique to discover the network performance prediction improvements by augmenting the data with identified events with potential impact.

The rest of the chapter is organized as follows: Section 4.1 explains the selection process of sports events and introduces the event data. Section 4.2 introduces the statistical test, presents the conducted experiments, and discussed the results. Section 4.3 details the proposed Deep learning solution for inspecting improvements in Network KPI predictions and describes the conducted experiment. Section 4.4 provides a conclusion to the chapter.

4.1 Sports Events

American Football, Baseball, and Basketball are the most popular sports in stated order in the USA [7] based on TV ratings. The most popular tournaments for these sports are the National Football League with 111 Million TV views, Major League Baseball with 40 Million TV views, and National Basketball Association League with 30 Million TV views since 2005. We identify the sport league schedules that intersect with the timespan of network data at hand. The Major Baseball League does not overlap with the network data timespan. The other two league schedules overlap with our network data's timespan; hence, we select the National Football League and National Basketball league. The events data for both leagues are extracted from the resource in [4].

SubSection 4.1.1 introduces the National Basketball Association League data and the required preprocessing steps. SubSection 4.1.2 presents the National Football League data and the appropriate preprocessing steps.

4.1.1 National Basketball Association League

The extracted data contains a row for each game played in the 2017-18 NBA session. Each row of the data holds information for a single league match of the NBA.
The dataset contains the following variables:

1. Date: Date of the game
2. Start(ET): Start time of the game
3. Visitor: Team that visits the home stadium of Opposite team
4. Home: Team that is playing at their home stadium
5. Attendance: Number of people who attended the game at the stadium

Data Filtering

According to industry experts at EXFO, the network parameters are affected by the unusual magnitude of packet traffic. We suspect that an NBA game impacts the network traffic around the match location and at places with the most fans (home cities of each team). Hence, an impact in network usage around the network sites might directly affect the network performance. In terms of NBA games, peoples' behaviour changes when their favourite team plays a game.
Therefore, we select the teams with a home city that contains at least one of the network monitoring sites. There are two such teams. To maintain anonymity of the proprietary network performance data, we address teams as team one and team two and the network sites associated with each team's home city as site one and site two in that order. For this

analysis, we select the games played by the chosen teams such that their dates coincide with the period of network data.

The Figure 4 presents a line plot for the total packet loss on one particular NBA game day and other non-game days

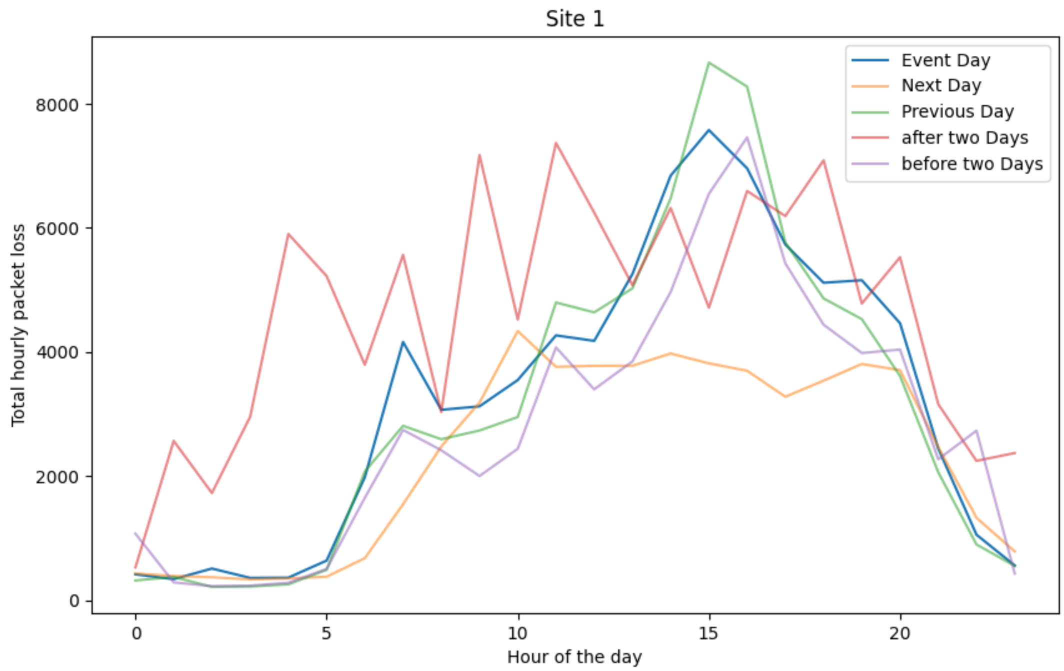

Figure 4: Hourly packet loss on different days at site 1

We observe that a direct visual inspection does not hint at any differences in the network performance on game day vs. non-game days. We perform a statistical analysis detailed in Section 4.2 to derive concrete inferences.

4.1.2 National Football League

The collected NFL events data contain follow important features:

1. Date: Date of the game

2. Time: Start time of the game

3. Loser/tie: Indicates the loser of the game

4. Winner/tie: Indicates the winner of the game

5. Attendance: Number of spectators in the stadium

Data Filtering

We use the same logic to filter NFL events as we applied to filter NBA events.

We observe that three cities are home to one of the NFL teams and consist of a network monitoring site. For anonymity, we address the teams and network sites as Team 1, Team 2, Team 3, and Site 1, Site 2, and Site 3 such that Team X's home city is the same as the location of Site X. Figure 5 shows the total hourly packet loss at Site 1 during an NFL event day and other non-event days.

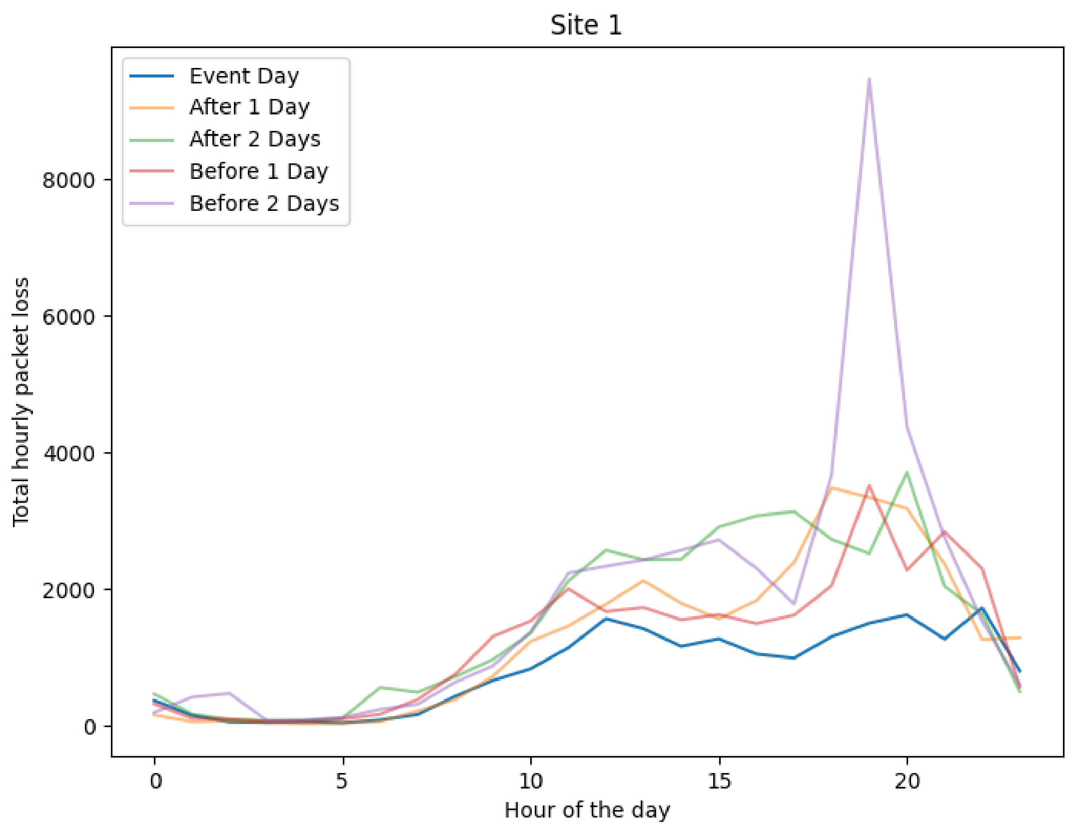

Figure 5: Hourly packet loss at site 1

4.2 Statistical Analysis

This section summarizes a statistical significance test to examine the network performance impact during sports games. This test is performed to examine the means of two time series to determine if they are statistically the same. The rest of the section introduces Welch's test, followed by the statistical tests and analysis.

4.2.1 Welch's Two sample t-Test

Welch's test is the two-sample t-test for testing the hypothesis that two populations have equal means. This test is more reliable when the samples have unequal variances, or unequal sizes [47]. In our case, the samples have an unequal size because each NBA team plays approximately 50 games from December to May (which overlap the network data time frame). For this analysis, we use an existing implementation of Welch's t-test from SciPy library [45].

4.2.2 Statistical analysis of NBA League and Network Parameters

We hypothesize that sports games impact the motive and total time of network usage, affecting network performance. Such an effect might be observed due to weekends and public holidays. Therefore, data points on US public holidays and weekends are removed from the network data to avoid detecting their effect on network traffic.

We observe that almost all the NBA games start between 7 pm - 8 pm. An average NBA game lasts 3 hours. However, the anticipation & interactions few hours before the game and post-game reactions and analyses might impact the network user behavior just like the actual game. Hence, we intuitively consider 4 pm to 12 am as hours of the potential impact and divide the data into following two parts.

1. Days with an event: This subset contains network data from 4 pm to 12 am on the relevant team's game days. We identify this partition as (A).

2. Days without an event: This subset contains network data from 4 pm to 12 am on days without the relevant team's game. We identify this partition as (B).

Results and Inferences

We define the following null hypothesis and perform the Welch's test to examine its correctness.

- Null Hypothesis: The means of samples (A) and (B) are equal.
- Alternate Hypothesis: The means of samples (A) and (B) are unequal.
- With a significance threshold of 0.05, Welch's test is performed using (A) and (B) for each site to test the hypothesis.

The resulting p-values for site 1 and site 2 are 0.69 and 0.08.

The results are summarized in the Figure 6.

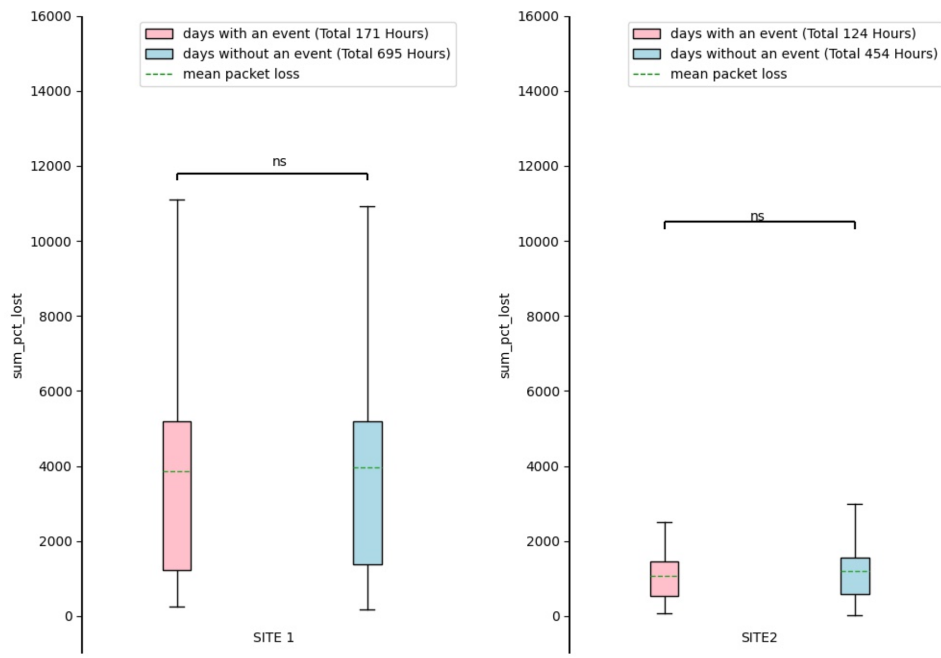

Figure 6: NBA: Box Plots with Welch's test results

These p-values indicate that the probability of sample means being different by chance and randomness is higher than the significance threshold. Here, we fail to reject the null hypothesis. This implies that the average network performance on an event day is not significantly different from the average network performance. The Network might be robust enough to maintain the performance and withstand potential impact during the NBA events. However, we do not expect the NBA event games to contribute to improving the network performance prediction as the network performance is not statistically impacted during the NBA games.

4.2.3 Statistical analysis of NFL and Network Parameters

NFL games have the highest average in-person attendance, and at-home views among all the sports in the United States of America [7]. As a result, more people across a broader territory might be affected by an NFL game. Hence, we use the games played by all three teams to perform a statistical test at each network site to examine the network performance impact during the games.

We observe that all NFL events are organized on a Sunday or a Monday. As a result, we analyze the network performance on NFL event days against the network performance on Sundays and Mondays without any NFL event. All NFL events occur in the afternoon, and each lasts for an average of three hours.

The audience at home engages with the event a few hours before the game while analyzing the previous games, anticipating the outcomes, and watching reruns of previous games. The attendees of an event tend to reach the stadium at the earliest to locate the best spot for match viewing. Many viewers tend to engage in pre-match and post-match activities at the venue and on the internet.

We expect an impact on network performance during such pre-event and post-event activities during the game days. Therefore, we consider the network performance from 12 PM to 12 AM (hours of potential impact) to identify the network performance impact during an NFL game.

We separate network performance data for the statistical analysis as below.

1. Days with an event: This subset contains network data during hours of the potential impact on game days. This dataset is addressed as (A) in the rest of the Section.

2. Days without an event: This subset contains network data during hours of the potential impact on Sundays and Mondays without any NFL event This dataset is addressed as (B) in the rest of the Section.

Results and Inferences

We use the datasets (A) and (B) to examine the impact of NFL events by performing the Welch's test. The hypotheses and significance threshold are defined as follows

- Null Hypothesis: The means of samples (A) and (B) are equal.
- Alternate Hypothesis: The means of samples (A) and (B) are unequal.
- Significance Threshold: A threshold of 0.05 is used to test the significance.

Performing multiple simultaneous hypothesis tests can increase the chances of observing a rare event, increasing the probability of incorrectly rejecting a null hypothesis [32] . Hence we use the Bonferroni correction to address the issue and obtain a new significance threshold following the equation 5.

$$\text{Corrected Threshold} = \text{Original Threshold}/\text{Number of statistical tests} \tag{5}$$

The corrected significance threshold for each test is 0.008.

Figure 7 presents the box-plot of samples for each network site with a dashed line to indicate the mean value of the samples. The Table 4 shows the p-values for corresponding to Welch's test for each network site.

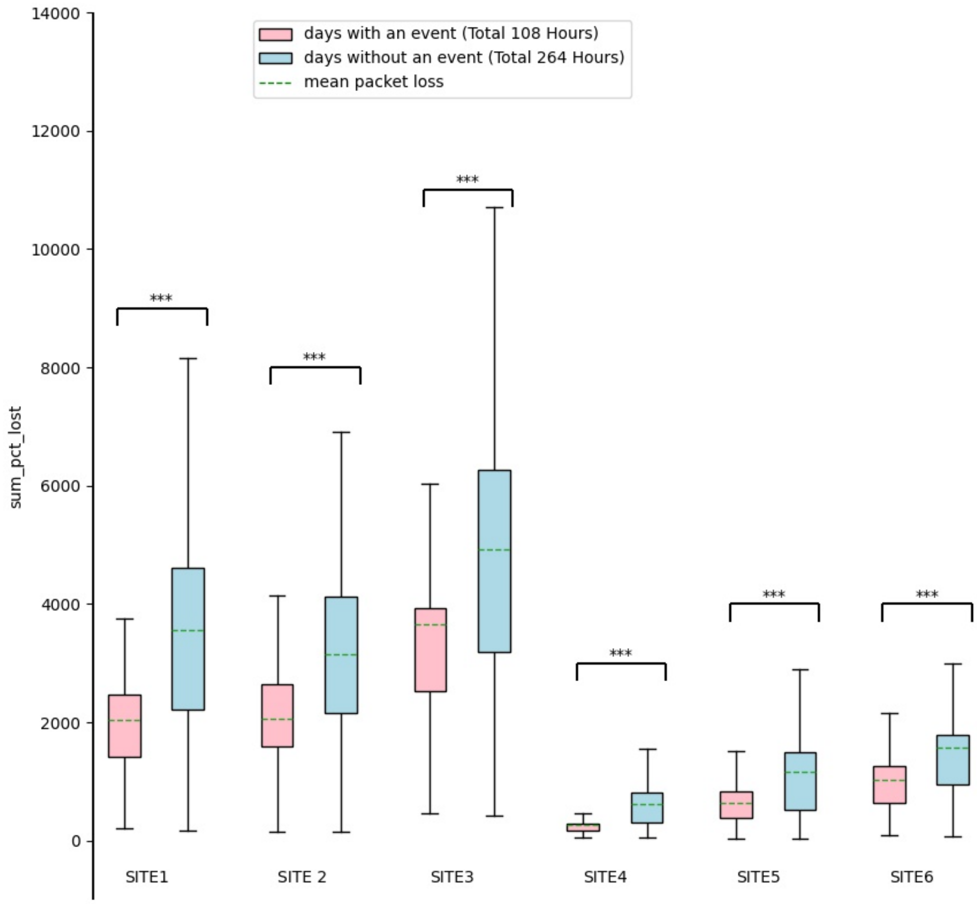

Figure 7: NFL: Box Plots with Welch's test results

Table 4: Welch's Test p-values

Site 1	Site 2	Site 3	Site 4	Site 5	Site 6
3.54×10^{-17}	1.62×10^{-15}	1.62×10^{-15}	6.34×10^{-21}	7.12×10^{-9}	6.06×10^{-8}

At each network site, we observe that the p-value of test is significantly below 0.008. Hence the null hypothesis is rejected to accept the alternate hypothesis implying that there is less than 1 percent probability that the sample means are different due to statistical chance and randomness.

This means that the average network performance in the subset (A) differs significantly from the average network performance in the subset (B).
Consequently, we deduce that the NFL events can an interesting factor that potentially impacts the network performance parameters. Hence, we augment the hourly packet loss data with NFL events to examine improvements in hourly packet loss prediction.

Performed statistical tests reflects improved network performance during an NFL game day. This is counter-intuitive to a certain extent as we expect the network load to increase due to increased online activity on an NFL game day. One probable explanation is that during an NFL game, people tend to watch the games in groups and reduce data usage from individual devices. Another possible reason is that network providers allocate more network resources around the event location during highly anticipated events, resulting in reduced overall network load, thus improving the network performance.

4.3 Network Parameter Predictive Analysis

We observe a difference in network performance during NFL events. As a result, We hypothesize that the network performance data can be augmented with the NFL events information to improve the network performance prediction. The rest of the section elaborates on the experiments to test our hypothesis.

4.3.1 Recurrent Neural Networks and Long Short Term Memory Units

Multiple studies show that recent advancements in Deep learning (RNNs) have enabled the Deep learning methods to outperform the conventional methods for the task of time series prediction. RNNs can efficiently learn the temporal information in a sequence because of recursive units that learn the model parameters from all the time steps in an input sequence. Due to this fact, RNNs have been used extensively in time series prediction tasks.

A plethora of external phenomena affect the network usage making the network data very complicated and non-linear. A highly anticipated sports game might affect the network for a few hours of the day, while conferences and online game launches can potentially disrupt the network behaviour for days. Modelling all such intricacies needs an architecture capable of handling and retaining information from a faraway temporal data sequence. RNNs are limited due to the vanishing and exploding gradients problems. They cannot efficiently retain information very far in the temporal sequence because of their architecture. Exploding gradients can be handled by clipping them to keep them from reaching infinity. Addressing vanishing gradients in RNNs is notoriously difficult and expensive.

Hochreiter and Schmidhuber [21] proposed the LSTM unit to address the vanishing gradient problem. An LSTM unit is more complicated than an RNN unit as the LSTM architecture has four trainable parameters for each RNN's trainable parameter. The architecture modification of the vanilla RNN addresses the vanishing gradient problem. Consequently, the solution to the vanishing problem comes with higher model complexity. Hence we use RNN with LSTM units for the rest of the experiments.

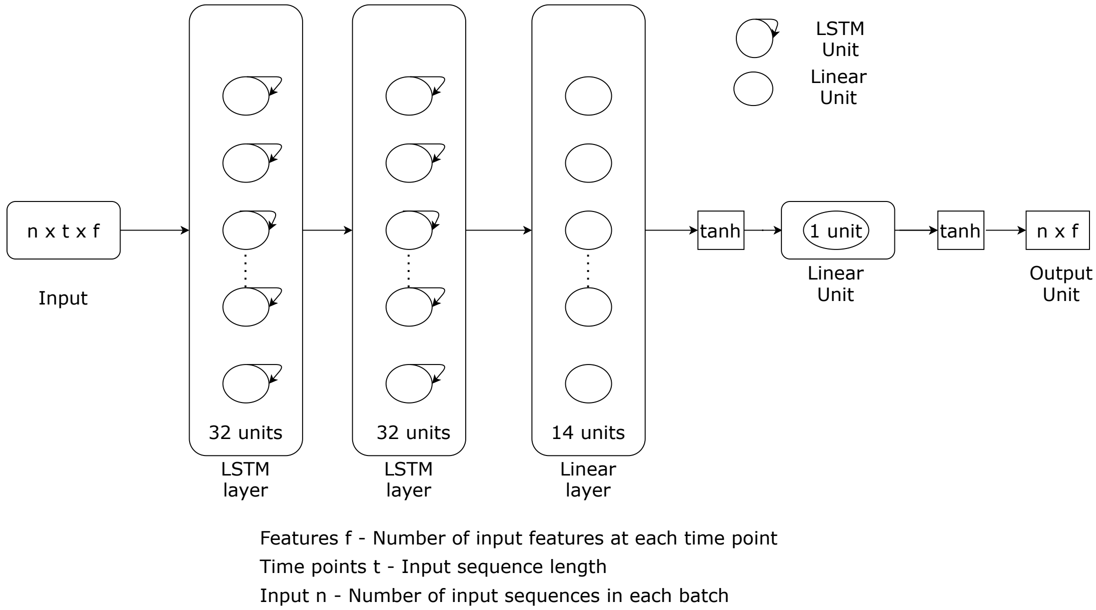

Figure 8: LSTM model architecture used for network performance prediction

4.3.2 Experiment Setup

We design a Deep learning Model consisting of RNNs with LSTM and Feed Forward networks to determine if the NFL event information can improve network performance prediction. Figure 8 provides an overview of the model architecture used in this research.

While the impact is observed at all the network sites, the most observable impact is expected at the network sites associated with the participating teams. We use the same architecture to train two types of models identified as Type 1 & Type 2 models with different input dimensions. We compare their prediction results and conduct the same experiments for all three network sites associated with an NFL team city.

- Type 1: Models are trained before network performance data is augmented by NFL events information.

- Type 2: Models are trained after augmenting the network performance data by NFL events information.

4.3.3 Dataset and Data Augmentation

This section describes the performed data augmentation and feature engineering to integrate NFL event information with the network packet loss data and then highlights the features used to train the two type of models. The hourly packet loss is a time series ranging from December 2017 to May 2018 with an hourly frequency. Each value is the total packets lost in that hour as shown in the Figure 1.

Converting discrete events into time series

A time series with values in $\{0,1\}$ is created with the hourly frequency and same length as the network performance variable "hourly packet loss." In such time series variables, the hours of interest are encoded as "1" and all the other hours are encoded as "0."

Feature Engineering

We define new features that can improve the performance predictions as follows:

The general population would transition from work-related network usage to personal network usage on weekends. We expect this behaviour change in users to affect the network load. Therefore, we create a cyclical feature called "Weekday" that encodes the day of the week by a unique numeric value. This feature represents the day of the week indicated by an integer in range [0,6]. "0" represents a Monday and "6" represents a Sunday. As the packet loss has hourly frequency, the "Weekday" value assumes identical value for 24 hours depicting a single day. The following features are defined to encode temporal information about NFL events.

- Site event: This is a binary time series variable that encodes the NFL gameplay. All the hours of NFL gameplay are represented as "1" and the other hours are encoded as "0."

- Site pre-event: This is a binary time series variable that encodes the hours before NFL gameplay. The three hours before all NFL games are encoded as "1" and the other hours are encoded as "0."

The number of data points indicating either an ongoing NFL event and the hours before an NFL event is comparatively lesser than the data points during no NFL gameplay.
A Deep learning model might consider the changes in value of such variables as noise as they might not provide additional information to predict network performance during non-game days, potentially leading the model to significantly rely on the network packet loss information.
Hence we add redundancy in terms of a new time series variable, "Site event and pre-event," as a possible strengthening mechanism so that the model might learn to utilize the event information more efficiently. We also limit our data till the third week of January since no relevant NFL games are played after that week.

- Site event and pre-event

  This time-series is the point-wise summation (feature cross) of the features "Site pre event" and "Site event".

We do not use a feature identifying post-event hours with the intuition that LSTMs can efficiently derive the post-event based on the event information because the information flows in the forward direction (Unidirectional LSTMs). We tested this intuition and did not observe degradation in model performance when the post-event information is excluded from training data.

We use the features "packet loss" and "Weekday" to train Type 1 models for network performance prediction. And we use the features "packet loss," "Weekday," "Site event," "Site pre-event", and "Site event and pre-event" to train Type 2 models for network performance prediction.

4.3.4 Cross Validation in Time Series Prediction

Proposed Cross Validation splits

As the dataset spans six weeks, we separate each week and propose a Cross Validation setup as shown in Table 5.

Table 5: Cross Validation - 5 splits

Training set(Week)	Validation set(Week)
1,2,3,4	5,6
2,3,4,5	1,6
3,4,5,6	1,2
2,4,5,6	1,3
1,3,5,6	2,4

Data Preparation

The model accepts fix-sized input sequences to produce the next time-point predictions. Faulty input sequences or targets can lead to inaccurate training and prediction results.

An input sequence is faulty if each value does not follow the previous one in time. For example, a 24 point input sequence consisting of 12 data points from week one and the other 12 points from week 4 data is a faulty input sequence. An input sequence-target pair is faulty if there is a time gap between the label time point and the last point of the input sequence. An example is an input sequence consisting last 24 points from week one and the first point of week three as the label is considered faulty.

Above described faults are possible while preparing the training and validation data for

Cross Validation splits that contain non-consecutive weeks of data (data with time gaps). Consequently, we generate sequences and their targets for each week separately and combine them later to produce different Cross Validation splits to train both types of models.

4.3.5 Experiments and Results

We perform the above-described CV for all the three network sites associated with an NFL team and compare the average mean absolute percentage prediction error for Type 1 and Type 2 models.

Set of Hyperparameters

We use one train-validate split to optimize the network hyperparameters to obtain baseline predictions experimentally for both the type of models. We utilize these hyperparameter values to train the models in the Cross Validation experiment. Experimentally, we get good baseline predictions using the following hyperparameter value.

- Input sequence length: 24

  The number of past data points used by the model to learn and predict the network performance for the next time-point. By this hyper-parameter, we control the direct amount of past information exposed to the model for a single prediction. Increasing the input sequence length increases the training time complexity and can improve model performance.

- Learning Rate: 0.001

  After processing each training batch, the model calculates the loss based on the predictions. After calculating the loss, the model takes controlled steps in the direction of minimum loss by updating the model parameters. This variable controls these steps.

- Number of Epochs: 1000

  This parameter represents the number of times the model goes through all the training data made available for the experiment.

- Number of LSTM layers: 2

  This parameter defines the number of hidden layers with LSTM units. Hermans and Schrauwen [19] found that using more than one hidden layer improves the intermediate representations learnt by hidden layers and ultimately helps the model achieve better performance. Hence increasing the number of such hidden layers can improve model performance.

- Number of LSTM units in each layer: 32

  This parameter represents the number of LSTM units in each LSTM layer. Increasing the number of LSTM units increases information retention since each unit has its memory. Hence, more LSTM units in a layer can contribute to better model performance. However, increasing the number of LSTM units and the number of LSTM layers in an irresponsible manner can increase the model complexity and result in over-fitting.

- Dropout Probability: 0.3

  This parameter introduces a dropout layer on top of LSTM layers, ignoring the layer output for each LSTM unit with pre-defined probability. This procedure reduces over-fitting chances by explicitly forgetting some currently learned information, forcing the model to engage other LSTM units to optimize the model.

Cross Validation Results and Inferences

We train Type 1 and Type 2 models for each Cross Validation split using the above-derived hyper-parameters. Model performance is represented using the Mean Absolute Averaged

Error metric. Table 6 presents the validation set model performance for all Cross Validation splits corresponding to all sites.

Table 6: Mean Averaged Percent Error across all event hours in the validation set

	Site 1		Site 2		Site 3	
CV folds	Type 1	Type 2	Type 1	Type 2	Type 1	Type 2
1	29	0.36	0.41	0.49	0.87	0.27
2	0.98	0.76	1.42	0.62	2.18	0.4
3	0.92	1.57	1.43	0.7	2.18	0.53
4	0.76	1	1.43	0.72	1.98	0.58
5	0.54	0.68	0.52	0.51	2.02	0.39
Average	6.46	0.874	1.042	0.604	1.846	0.434
STD DEV	12.6	0.45	0.53	0.11	0.55	0.12

We observe that type 2 models outperform type 1 models consistently across most Cross Validation splits corresponding to Site 2 and Site 3. This observation is not valid for models corresponding to Site 1.

For Type 1 models corresponding to Site 1, we observe a high variability in model performance across five Cross Validation splits. One potential reason might be that the network performance at Network Site 1 is more convoluted than the other two. Multiple external factors might affect the network performance at a given time-point resulting in a very intricate and complex variable that is more difficult to model than the network performances at the other two Network sites. Interestingly, Network Site 1 is located around the biggest and most populous city among them. This potentially imply more network users accessing network for distinct reasons affecting the network performance in multiple ways.

However, we observe significant prediction improvements for Cross Validation splits at Site 2 and Site 3. The error values exhibit less variability. These results support our hypothesis that NFL event information contributes to improving the network performance prediction.

Now that we have experimental evidence of prediction improvements by augmenting performance data by NFL events, we optimize the model hyper-parameters to obtain an optimized network performance prediction model for each Network Site. We detail this experiment and quantify the prediction improvements in the rest of this chapter.

4.3.6 Model Optimization by Automated Hyperparameter Tuning

We established that network data augmentation by NFL events improves the network performance predictions. A company can execute better business decisions using our models' predictions if they are as accurate as possible. Hence, we try to optimize the prediction models at each network site by exploring different hyperparameter combinations. We present an experiment to evaluate different combinations of hyper-parameters using the library Ray-Tune [30] to enhance model performance.

Hyperparameter Search Space

The library provides multiple data structures to define the hyperparameter search space. The parameter selection and the number of hyperparameter combinations used for optimization depend on the type of data structure used for search space definition. We describe the structures used for our experiments.

- tune.grid_search

  Forces the algorithm to evaluate all hyperparameter combinations with each value defined for this variable.

- tune.choice

  Provides a value (with equal probability) from a set of discrete options defined using this variable.

- tune.uniform

  Provides a range of continuous values from which a value is chosen with uniform distribution (equal probability)

The library also provides a scheduler mechanism to prune the under-performing models and intelligently select the hyperparameters based on previous performance improvements.

Dataset

We use the dataset described in SubSection 4.3.3. Now that we want to improve the prediction performance of our model, the Cross Validation approach is not used in this experiment. We define a single train-validate-test split for the rest of the section. The training data spans four weeks, while the validation and test data span one week each.

The Experiment

Book [23] states that the range of values for each hyper-parameter must be defined based on previous experience and initial experiments. We define an initial search space with the help of hyper-parameters obtained from the cross validation experiment in SubSection 4.3.5. Then we expand and modify the initial search space to further optimize the model performance. Table 7 shows the initial search space and Table 8 depicts the modified search space.

Table 7: Initial search space

epochs	tune.grid_search[500,700,1000]
n_steps	tune.grid_search[48,72,84]
hidden_layer_size	tune.grid_search[32,64,128]
num_layer	tune.grid_search[1,2,3,4,5]
learning rate	tune.grid_search[0.001, 0.0005, 0.0007, 0.0001]

Table 8: Updated search space

epochs	tune.choice[500,700,1000,2000]
n_steps	tune.uniform[24,84]
batch_size	tune.uniform[8,16]
hidden_layer_size	tune.choice[8,10,12,16,24,32,48,64]
num_layer	tune.choice[1,2,3,4]
learning rate	tune.choice[0.0001, 0.0005, 0.0007, 0.001, 0.0025, 0.005, 0.01]

We specify distributions to sample hyper-parameter values in updated search space to explore more hyper-parameter combinations. Two thousand different combinations of hyperparameters are sampled following specified distributions from the updated search space to find the optimized model.

To obtain a performance comparison of optimized models, we use both search spaces to separately optimize Type 1 and Type 2 models. All the models are trained on the training dataset. We select the models that provide the highest prediction accuracy on the validation dataset. Then we measure the prediction performance on the test sets to inspect model performance. Table 9 represents the model performance on the validation set, and Table 10 represents the model performance on the test set using the best model configuration provided by the experiment from each type of model.

Table 9: Mean Averaged Percentage Error on validation set

Site	Type 1	Type 2
1	0.43	0.38
2	0.59	0.41
3	0.63	0.5

Table 10: Mean Averaged Percentage Error on test set

Site	Type 1	Type 2
1	3.36	0.36
2	0.59	0.4
3	0.67	0.51

We observe a Type 2 model prediction accuracy improvement by at least 23 % on test data at each network site compared to the respective Type 1 models.

4.4 Discussion and Conclusion

Figure 7 shows that the network performance during NFL events significantly differs from the rest of the network performance. Moreover, the LSTM predictions show that data augmentation with NFL events significantly improves the predictions. Owing to the prediction improvements, we infer that NFL events have a significant impact on network performance. However, the NBA events do not incur a similar impact on the network performance as observed in Figure 6.

As mentioned in Section 4.1, NFL is the most popular sports league in the US. It is vastly more popular than the NBA league regarding stadium attendance and at-home views [7]. Popularity and viewership might be an essential distinguishing factor to measure any events' impact on the network performance and their potential to improve the network performance predictions.

Chapter 5

Tweets for improved Network KPI predictability

In the last decade, major news, sports, and government organizations have started to publish news on Twitter. Twitter allows the users to instantly react to news and events by publishing messages called tweets. Twitter data has been extensively used to perform various tasks. Vieweg et al. [44] used posts generated during two disaster events to identify and extract information to facilitate disaster response and management. Blanford et al. [12] used peoples geo-tagged tweets and their temporal information to map regional and cross-border connectivity in Kenya. A 2019 Forbes article in [28] studied Twitter, news, and Google search activities during multiple disasters and concluded that while news media provide the earliest warnings, Twitter activity rises during an active event. The study suggests that Twitter data should be perceived as a behavioural and attention signal to understand event insights. Martín et al. [31] performed Twitter messages analysis to evaluate and locate the activities in the city of Valencia to improve special event management. Twitter activity is influenced by global and local events. As mentioned earlier, special events can help in estimating their impact on a telecom network performance. However, keeping track of all local activities becomes expensive as the services expand geographically.

Twitter is a very important social network and information network [33]. The Telecom network performance is influenced by the active number of users and type of access. The increase in Twitter activity suggests an increase in the number of people engaging on the platform. Depending on the reasons for such activity increase, the number of active network users, type of network usage can be potentially affected.

Owing to the above discussion, we use local Twitter activity data at each network site to inspect improvements in the (site-specific) network performance prediction. We perform a statistical correlation analysis between Twitter and network performance data to discover similarities between these variables and describe the experiments to analyze prediction improvements of network performance using Twitter data.

The rest of the chapter is organized as follows: Section 5.1 introduces the collected Twitter dataset and preprocessing performed for the experiments. Section 5.2 defines the correlation experiment and results. Section 5.3 presents the Deep learning experiment, results, their statistical significance, and conclusion.

5.1 Twitter Data

This section details the process of tweets collection.

5.1.1 Data Collection

We collect all the tweets posted on Twitter between December 2017 and May 2018 within a circular area with a 50 km radius measured from each network site. We use the open-source library Twint [8] to enforce the geographical constraint and collect the tweets. In total, six Twitter data files are generated that contain tweet features around a network site. The collected data includes following important fields:

1. Date: Date of the tweet

2. Day: Day of the week

3. Tweet: The whole tweet

4. Place: Geo-location of the tweet

5.1.2 Transformations on the acquired data

We want to correlate the Twitter activity in a region around each network site with its network performance. However, there is no predefined limit around the network sites to limit the tweets collection for analysis. Consequently, we define multiple boundaries to filter the tweets at each network site.

At each network site, the tweets are filtered from a circular area around the network site based on the radii: 15km, 25km, 40km, 50km and stored in CSV files separately. To capture the changes in Twitter usage, a feature "tweet count" is calculated using the filtered tweets.

- Tweet Counts: Frequency of tweets every hour

The correlation of filtered tweet counts and network performance data is calculated at each site. Then we perform comparative correlation analysis to detect if the size of the geological area used to filter tweets play any significant role in the resulting correlation.

Table 11 summarizes the total, average, and median tweet count per hour when filtered with a 50 km radius at each site. We observe that the network sites located around small cities have very few tweets every hour, and larger cities have more average tweet counts every hour.

Table 11: Summary of tweets volume

	Total tweets across months	Average count per hour	Median count per hour
Site1	234 150	64	72
Site2	94 163	25	27
Site3	15 042	4	4
Site4	435 172	119	137
Site5	216 795	59	66
Site6	44 376	12	12

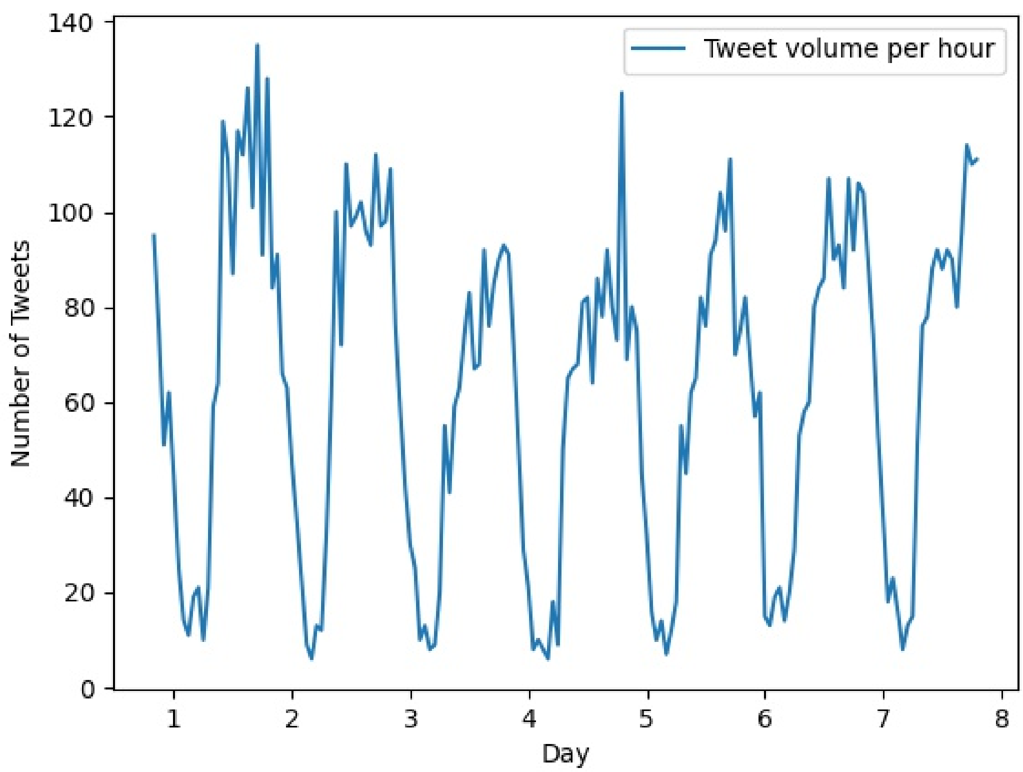

Figure 9: 7 days of tweet volume per hour

5.1.3 Seasonality Adjustment

The Figure 9 shows a plot of tweet volume every hour at one of the sites for seven consecutive days. We observe a cycle that repeats every day. The feature assumes a higher value as the day progresses and then drops as dusk sets in. We expect this cycle because most users tend to sleep at night and remain the most active during the daytime. We observe a very high correlation between tweet volumes and packet loss as their values follow a 24-hour pattern observed in both the time series, as outlined in the Figure 10. However, we want to examine the underlying similarities in the features that can improve network performance predictability. Hence, we perform the seasonality adjustment of Twitter data using the Seasonal Decompose method discussed in Section 3.2. Figure 11 shows the decomposed tweet volume per hour corresponding to one of the network sites.

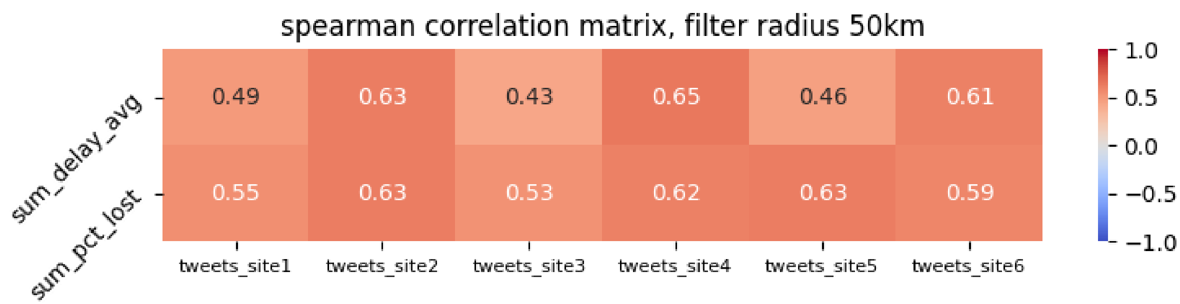

Figure 10: Correlation before Seasonal Decomposition

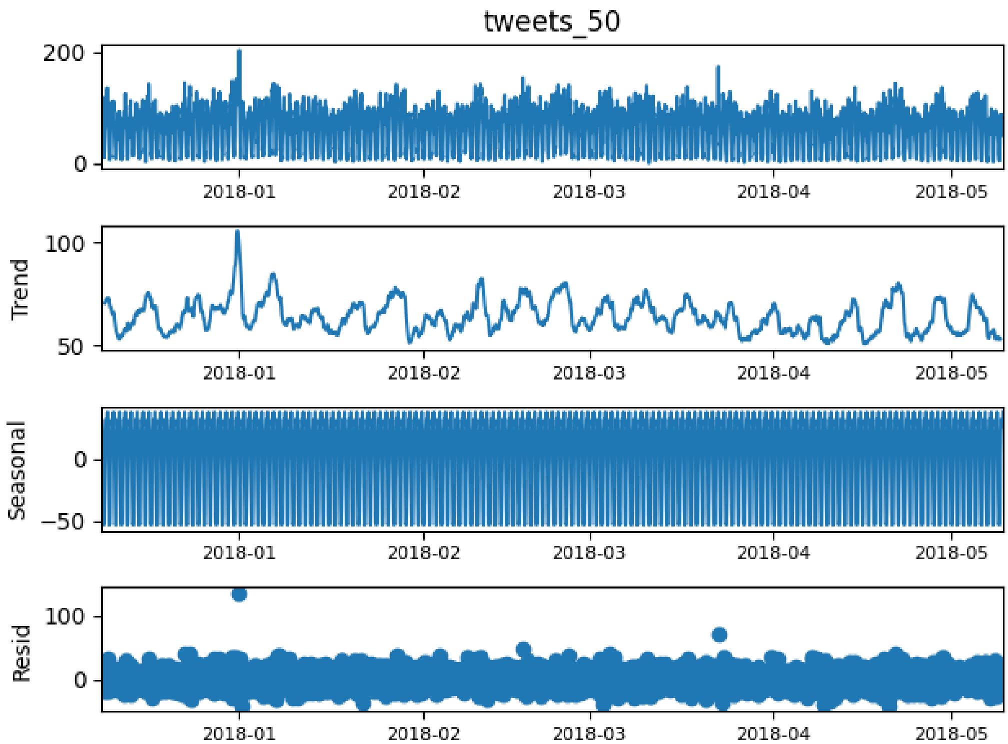

Figure 11: Decomposed Twitter variable associated to one network site

5.2 Correlation Analysis

We use the Spearman correlation summarized in Section 3.3 to generate correlation results in this analysis.

The trend component of the packet loss feature calculated in Section 3.2 and the trend component of the tweet volumes filtered with different radii are used to calculate the correlation coefficient for each network site. The correlation results are summarized in the Figure 12.

5.2.1 Inferences and Discussions

Based on the results at each site, we do not observe a consistent change in the magnitude of the correlation coefficient while moving from a smaller to a larger Twitter data filter radius.

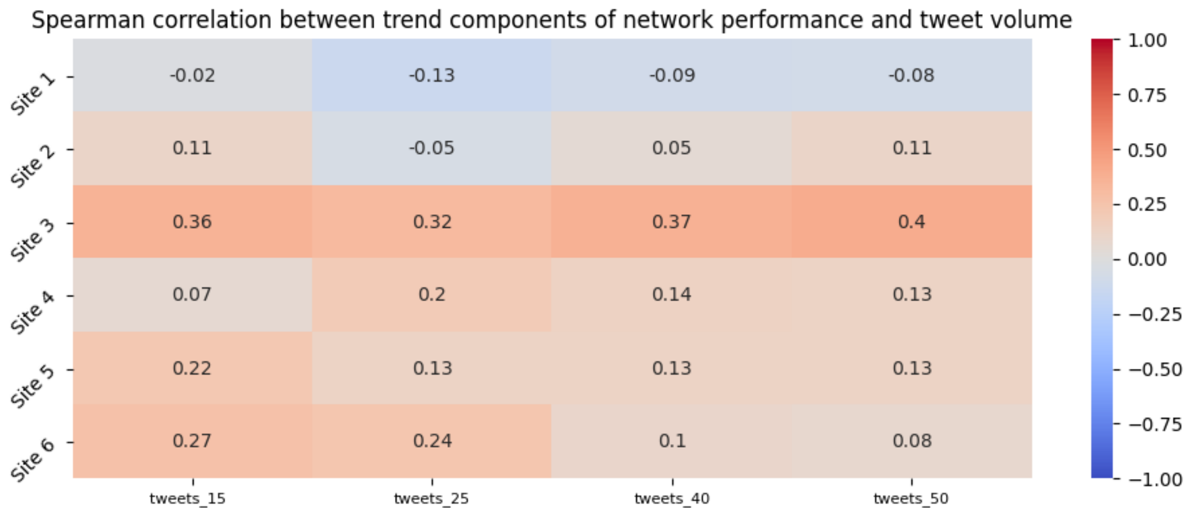

Figure 12: Spearman correlation between trend components of packet loss vs tweets volume

The only exception to this observation is the correlation of packet loss vs. tweet volumes at Network site 6 with a consistent decrease in correlation magnitude.

Figure 12 shows an average correlation of 15 percent or more at each site other than site 1 and site 2. As Twitter activity relies on network services, an increase in active Twitter usage can reflect an increase in the network usage and network load. Any planned event might cause an increase in the number of people streaming it on the network and expressing their views and commentary on Twitter at the same time, which can add to the correlation of both time series. Local news and activities that concern the nearby population, like an unexpected storm, can lead to the spread of information and caution via twitter resulting in increased network access.

The correlation results confirm the existence of significant similarity between the trends of tweet volumes and packet loss. This implicates that the underlying factors affecting the network performance might have a similar effect on the tweet volumes (Positive correlation). Hence, the prediction mechanism should benefit by augmenting the network data with tweet volume values. In the next section, we utilized the Neural Network, described in Section 4.3, to quantify the improvements in network performance prediction by augmenting the network data with tweet count information.

5.3 Network Performance Predictive Analysis

We train two types of models as defined in Section 4.3 to predict the network performance parameter. Type 1 models are trained without the Twitter data, and Type 2 models are trained after data augmentation using the Twitter data.

5.3.1 Walk Forward Validation

Walk Forward validation is a specific application of the Cross Validation technique [48]. The Walk Forward validation method defines multiple training sets with incremental size where each test set follows the corresponding training set. Having multiple train and test sets reduces model over-fitting [48]. Each test set temporally follows the corresponding train set. We define a five-fold Walk Forward validation split as shown in Table 12.

Table 12: Walk forward - 5 splits

Split	Training set(Month)	Validation set(Month)
1	1	2
2	1,2	3
3	1,2,3	4
4	1,2,3,4	5
5	1,2,3,4,5	Last 10 days

5.3.2 Experiment and Results

We define the following two types of models for the next hour performance prediction and inspect the improvements in the prediction of network performance.

- Type 1 models are trained without Twitter data using "Packet loss" and "Weekday" features.
- Type 2 models are trained with the additional feature "Tweet Counts".

The validation set results for each network site and walk forward validation split is provided in Table 13.

Table 13: 5 Split walk forward validation Mean Absolute Percentage Errors

	Site 1		Site 2		Site 3		Site 4		Site 5		Site 6	
	Type I	Type II	Type I	Type II	Type I	Type II	Type I	Type II	Type I	Type II	Type I	Type II
Split 1	0.81	0.73	1.09	0.82	0.68	0.43	1.15	0.9	0.68	0.62	0.81	0.67
Split 2	0.4	0.41	1	0.84	0.71	1.01	0.78	0.96	2.67	1.95	0.43	0.44
Split 3	0.77	0.75	0.84	0.52	0.29	0.42	1.2	1.17	2.98	2.24	0.72	0.65
Split 4	0.49	0.48	0.61	0.63	0.3	0.39	0.77	0.97	2.01	1.62	0.47	0.48
Split 5	0.64	0.6	0.43	0.47	0.25	0.23	1.2	1.2	0.58	0.58	0.68	0.63
Average	0.622	0.594	0.794	0.656	0.446	0.496	1.02	1.04	1.784	1.402	0.622	0.574
STD DEV	0.178	0.149	0.273	0.169	0.228	0.298	0.224	0.135	1.110	0.764	0.164	0.105

The error values across each split exhibit minor variability except for site 5, where the standard deviation is much higher than the other sites. Type 2 models improve the validation performance by 3% at site 1, 6% at site 2, 5% at site 6, and by a bigger margin at site 5. However, such improvements might reflect the randomness in the model training rather than deterministic improvement due to data augmentation using tweet volumes.

To determine if the average Type 1 models' validation set errors are statistically different from Type 2 models' validation set errors, we perform a significance test described as follows.

5.3.3 Significance Test

We compare the average errors (Type 1 vs. Type 2) at each site using a Student's t-test [42] and determine if the average Type 1 validation errors are statistically the same as Type 2 validation errors for each site. We define the null hypothesis, alternate hypothesis, and the significance threshold for the t-tests as follows.

- Null Hypothesis: The mean error of type 1 models and type 2 models are statistically the same.
- Alternative Hypothesis: The mean error of type 1 models and type 2 models are statistically different.
- Significance Threshold: We use a significance threshold of 0.05 (5%).
- Bonferroni corrected Threshold: 0.008

Table 14 denotes the p-values associated with the test performed at each site. We observe a p-value of 0.15 or higher at all sites, which indicates a minimum 15 percent chance that the observed difference in error values is statistically random and not statistically different. Hence, we fail to reject the null hypothesis of performed t-tests between walk-forward performances across all sites and derive that the network performance data augmentation using tweet volumes does not contribute to improving network performance prediction using the proposed LSTM model.

Table 14: t-test P-values for each network site

	Site 1	Site 2	Site 3	Site 4	Site 5	site 6
P-value	0.396	0.182	0.386	0.434	0.272	0.299

The Type I error is the possibility of falsely rejecting the null hypothesis due to the erroneous p-value of the test. Nadeau and Bengio [34] examined the type I and type II errors associated with the t-tests applied on various types of Cross Validation results to compare two models. They point out that the p-values might erroneously dip towards significance due to violation of the data independence assumption, potentially resulting in high type 1 errors. Such a violation does not affect the current analysis as we do not reject the null hypothesis for any performed t-tests. Based on the above statistical test results, we do not observe a significant prediction improvement after augmenting the packet loss data using tweet volumes.

5.4 Discussion and Conclusion

Due to trend correlations, we hypothesised that the behavior of Twitter activity might reflect the network performance behaviour. This motivated the network data augmentation and Deep learning experiment to inspect the packet loss prediction improvements. Insignificant improvements are observed in network performance prediction at few network sites. Hence, we conclude that the tweet volume data does not improve the packet loss predictions.

Performing a similar analysis for more network sites and broader geographical area might produce interesting results. The network performance data spans five months. Using a longer data span can also provide more walk-forward splits to compare the prediction performances and solidify our inferences. However, we performed an extensive analysis with network data at six network sites and expect similar results if this work inspires a deeper study.

Chapter 6

Conclusion and Future Work

In this thesis, we presented separate experiments to evaluate the impact of three different types of external factors on network performance. We performed exploratory data analyses using these factors and network performance KPIs to identify correlations between them. Finally, we performed the network packet loss data augmentation using external data, trained LSTM models on the augmented datasets, and measured the performance improvements resulting from data augmentation.

Data augmentation using NFL events improved LSTM prediction performance by more than 23 % at each considered network site. Trends in the temperature and wind speed values did not exhibit sufficient correlation with the network KPIs. Data augmentation using tweet counts did not provide prediction improvements at few sites and provided statistically insignificant prediction improvements at other network sites.

We identify the following future research possibilities. Extreme weather has the potential to impact various daily activities. Such weather events can also disable physical network devices, rerouting and increasing network traffic on other locations. Certain weather events (heat waves, freezing temperatures) might prompt people to remain home and increase the network load in the residential areas. Identifying and exploring such upcoming weather events and anomalies might improve network performance predictions and understanding.

Regarding sports events, event popularity and average viewership might help identify potential impact on the network performance. Hence, such event characteristics can be integrated into the exploratory analysis to identify events and to examine their impact on network. This work can also be extended to include other location-specific popular events including, but not limited to, political speeches and highly anticipated concerts. In addition, since service providers deploy additional resources to attenuate performance impact due to certain events, identifying the events which go unnoticed to the service providers may improve the network performance understanding and predictability.

Regarding social media, companies announce their products and exciting news on their Twitter handle to reach more audience. Depending on the nature of such announcements and news, network load might increase, resulting in performance degradation. Performing semantic analysis of tweet contents can also identify user activities with potential network impact. Such announcements and semantic information can be used to predict and understand the network KPIs. Finally, this research can also be extended using more prolonged data periods to analyze and quantify the network impact due to any of the above mentioned external factors.

Bibliography

[1] 2018 Ratings Wrap: NFL Laps the Field, Again. https://www.sportsmediawatch.com/2019/01/top-sports-audiences-2018-list/. (accessed: 2020-05-30).

[2] Frontier weather - dtn. https://www.dtn.com/weather/utilities/frontier-weather/?utm_campaign=frontierweather. (accessed: 2020-05-05).

[3] Percentage of households with internet use in the united states from 1997 to 2019. https://www.statista.com/statistics/189349/us-households-home-internet-connection-subscription. (accessed: 2020-05-20).

[4] Sports reference | sports stats, fast, easy, and up-to-date. https://www.sports-reference.com/. (accessed: 2020-06-07).

[5] Sports Tourism: State of the Industry Report. https://www.sportseta.org/portals/sportscommissions/Documents/Reports/TourismEconomics%20-%20Sports%20ETA%20SOTI%20-%20FINAL_82620.pdf. (accessed: 2020-05-31).

[6] statistical models, hypothesis tests, and data exploration. https://www.statsmodels.org/stable/index.html. (accessed: 2020-05-17).

[7] Top 10 most popular sports in america 2021 (tv ratings). https://sportsshow.net/most-popular-sports-in-america/. (accessed: 2020-06-07).

[8] Twint - twitter intelligence tool. https://github.com/twintproject/twint#readme, 2018.

[9] A. K. Ahmad, A. Jafar, and K. Aljoumaa. Customer churn prediction in telecom using machine learning and social network analysis in big data platform. *CoRR*, abs/1904.00690, 2019.

[10] M. Azeem, M. Usman, and A. C. M. Fong. A churn prediction model for prepaid customers in telecom using fuzzy classifiers. *Telecommunication Systems: Modelling, Analysis, Design and Management*, 66(4):603–614, December 2017.

[11] A. Bhorkar, K. Zhang, and J. Wang. Deepauto: A hierarchical deep learning framework for real-time prediction in cellular networks, 2019.

[12] J. Blanford, Z. Huang, A. Savelyev, and A. MacEachren. Geo-located tweets. enhancing mobility maps and capturing cross-border movement. *PLoS One*, 10(6), June 2015. Publisher Copyright: © 2015 Blanford et al. This is an open access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

[13] E. Brynjolfsson, L. M. Hitt, and H. H. Kim. Strength in numbers: How does data-driven decisionmaking affect firm performance? *SSRN Electronic Journal*, Apr 2011.

[14] C. Charlebois and J. Stevens. MORE THAN MONEY: Leveraging the Benefits of Sport Hosting in Niagara. Policy Brief 30, Brock University, Niagara Community Observatory, Centre for Sport Capacity. (accessed: 2020-06-05).

[15] A. Cheema and V. M. Patrick. Influence of warm versus cool temperatures on consumer choice: A resource depletion account. *Journal of Marketing Research*, 49(6):984–995, 2012.

[16] C. Gratton, S. Shibli, and R. Coleman. The economic impact of major sports events: A review of ten events in the uk. *The Sociological Review*, 54(2_suppl):41–58, 2006.

[17] J. He, W. Shen, P. Divakaruni, L. Wynter, and R. Lawrence. Improving traffic prediction with tweet semantics. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, IJCAI '13, page 1387–1393. AAAI Press, 2013.

[18] A. Hébert, T. Guédon, T. Glatard, and B. Jaumard. High-resolution road vehicle collision prediction for the city of montreal. *CoRR*, abs/1905.08770, 2019.

[19] M. Hermans and B. Schrauwen. Training and analysing deep recurrent neural networks. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013.

[20] A. Heyes and S. Saberian. Temperature and decisions: Evidence from 207,000 court cases. *American Economic Journal: Applied Economics*, 11(2):238–65, April 2019.

[21] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[22] R. Hofstede, I. Drago, G. C. M. Moura, and A. Pras. Carrier ethernet oam: An overview and comparison to ip oam. In *Proceedings of the 5th International Conference on Autonomous Infrastructure, Management, and Security: Managing the Dynamics of Networks and Services*, AIMS'11, page 112–123, Berlin, Heidelberg, 2011. Springer-Verlag.

[23] G. Ian, B. Yoshua, and A. Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[24] A. S. Khatouni, F. Soro, and D. Giordano. A machine learning application for latency prediction in operational 4g networks. In *2019 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, pages 71–74, 2019.

[25] A. Klein, C. Craun, and R. S. Lee. Airport delay prediction using weather-impacted traffic index (witi) model. In *29th Digital Avionics Systems Conference*, pages 2.B.1–1–2.B.1–13, 2010.

[26] A. Koesdwiady, R. Soua, and F. Karray. Improving traffic flow prediction with weather information in connected cars: A deep learning approach. *IEEE Transactions on Vehicular Technology*, 65(12):9508–9517, 2016.

[27] J. Krumm, A. L. Kun, and P. Varsányi. Tweetcount: Urban insights by counting tweets. In *Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers*, UbiComp '17, page 403–411, New York, NY, USA, 2017. Association for Computing Machinery.

[28] K. Leetaru. Is twitter really faster than the news? *Forbes*, Feb 2019.

[29] S. Li, A. Blake, and R. Thomas. Modelling the economic impact of sports events: The case of the beijing olympics. *Economic Modelling*, 30:235–244, 2013.

[30] R. Liaw, E. Liang, R. Nishihara, P. Moritz, J. E. Gonzalez, and I. Stoica. Tune: A research platform for distributed model selection and training. *arXiv preprint arXiv:1807.05118*, 2018.

[31] A. Martín, A. B. A. Julián, and F. Cos-Gayón. Analysis of twitter messages using big data tools to evaluate and locate the activity in the city of valencia (spain). *Cities*, 86:37–50, 2019.

[32] R. G. Miller. *Simultaneous statistical inference*. Springer, 1981.

[33] S. A. Myers, A. Sharma, P. Gupta, and J. Lin. Information network or social network? the structure of the twitter follow graph. In *Proceedings of the 23rd International Conference on World Wide Web*, WWW '14 Companion, page 493–498, New York, NY, USA, 2014. Association for Computing Machinery.

[34] C. Nadeau and Y. Bengio. Inference for the generalization error. *Mach. Learn.*, 52(3):239–281, Sept. 2003.

[35] A. Nikravesh, D. R. Choffnes, E. Katz-Bassett, Z. M. Mao, and M. Welsh. Mobile network performance from user devices: A longitudinal, multidimensional analysis. In M. Faloutsos and A. Kuzmanovic, editors, *Passive and Active Measurement*, pages 12–22, Cham, 2014. Springer International Publishing.

[36] J. Pamina, J. Beschi Raja, S. Sam Peter, S. Soundarya, S. Sathya Bama, and M. S. Sruthi. Inferring machine learning based parameter estimation for telecom churn prediction. In S. Smys, J. M. R. S. Tavares, V. E. Balas, and A. M. Iliyasu, editors, *Computational Vision and Bio-Inspired Computing*, pages 257–267, Cham, 2020. Springer International Publishing.

[37] L. Pierucci, A. Romoli, R. Fantacci, and D. Micheli. An optimized neural network for monitoring key performance indicators in hsdpa. In *21st Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, pages 2041–2045, 2010.

[38] M.-T. Puth, M. Neuhäuser, and G. D. Ruxton. Effective use of spearman's and kendall's correlation coefficients for association between two measured traits. *Animal Behaviour*, 102:77–84, 2015.

[39] D. Raca, A. H. Zahran, C. Sreenan, R. Sinha, E. Halepovic, R. Jana, and V. Gopalakrishnan. Back to the future: Throughput prediction for cellular networks using radio kpis. In *HotWireless '17*, 2017.

[40] S. Ranjan, S. Sood, and V. Verma. Twitter sentiment analysis of real-time customer experience feedback for predicting growth of indian telecom companies. In *2018 4th International Conference on Computing Sciences (ICCS)*, pages 166–174, 2018.

[41] Z. Sayeed, E. Grinshpun, D. Faucher, and S. Sharma. Long-term application-level wireless link quality prediction. In *2015 36th IEEE Sarnoff Symposium*, pages 40–45, 2015.

[42] Student. The probable error of a mean. *Biometrika*, 6(1):1–25, 1908.

[43] R. Toledo. The average household's internet data usage has jumped 38x in 10 years. `https://decisiondata.org/news/report-the-average-households-internet-data-usage-has-jumped-38x-in-10-years/`. (accessed: 2020-05-05).

[44] S. Vieweg, A. L. Hughes, K. Starbird, and L. Palen. Microblogging during two natural hazards events: What twitter may contribute to situational awareness. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, page 1079–1088, New York, NY, USA, 2010. Association for Computing Machinery.

[45] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.

[46] X. Wang, M. S. Gerber, and D. E. Brown. Automatic crime prediction using events extracted from twitter posts. In S. J. Yang, A. M. Greenberg, and M. Endsley, editors, *Social Computing, Behavioral - Cultural Modeling and Prediction*, pages 231–238, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[47] B. L. Welch. The significance of the difference between two means when the population variances are unequal. *Biometrika*, 29(3/4):350–362, 1938.

[48] Wikipedia. Walk forward optimization. `https://en.wikipedia.org/wiki/Walk_forward_optimization`. (accessed: 2020-07-10.

[49] Y. Yang, Z. Wei, Q. Chen, and L. Wu. Using external knowledge for financial event prediction based on graph neural networks. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, CIKM '19, page 2161–2164, New York, NY, USA, 2019. Association for Computing Machinery.