Final Report

Studio Session: Studio 1-3
Group Number: 6
Theme Number: 5

Group Members:
1. Parshwa Shah (Student ID: 104323824)
2. Lau Chong Zheng (Student ID: 104476139)
3. Vihanga Bandara (Student ID: 103833681)
4. Lucas Berry (Student ID: 103876288)

Contents

1. Introduction

1.1 Background and Motivation

5G networks are the newest generation of mobile internet technology, promising faster speeds and better connections than ever before. However, 5G performance isn't the same everywhere – it can work better in some places and worse in others. This difference in performance is a big challenge for companies that provide 5G services.

Our project has three main motivations:

First, we need better ways to understand why 5G performance varies in different places. Outdated methods for checking performance are insufficient. Using machine learning, we can better understand these patterns and help improve service quality. This is important because network companies need to know where and why their service isn't working as well as expected.

Second, 5G networks create lots of data about how well they're working. This data is valuable because it can help prevent problems before they happen. Instead of fixing issues after customers complain, network companies can spot potential problems early and fix them. This is why we're using machine learning to analyze the data and make predictions about network performance.

Third, cities have many different types of areas – some with tall buildings, others with open spaces, and some with lots of people using the network at once. Each of these areas can affect how well 5G works. Our project aims to understand these differences so that network companies can provide good service everywhere in the city.

Our work focuses on two main tasks:

- Grouping similar areas together based on their network performance
- Predicting how well the network will perform in the future

This project can help solve real problems in several ways:

- Finding the best places to put new 5G equipment
- Making sure busy areas have enough network capacity
- Fixing problems before they affect users
- Helping companies plan where to expand their network
- Making sure everyone gets good quality internet service

These solutions are especially useful for:

- Engineers who build and maintain 5G networks
- Companies that provide 5G services
- City planners who need to understand where network improvements are needed

By using data and machine learning to understand and predict 5G performance, we can help make mobile internet better and more reliable for everyone.

1.2 Project Objectives

Our project has two primary objectives:

1. **Group Geographical Zones Based on 5G Network Performance**
   Using clustering methods, we intend to detect and categorize various zones based on 5G performance characteristics like throughput and latency. This classification can help address issues such as:
   a. How many categories may be formed based on 5G network performance?
   b. What features distinguish each group in terms of network quality?

By doing this, our initiative offers insights into regions with unique network capabilities, facilitating the strategic development of 5G infrastructure and the efficient use of resources. The helps telecom engineers and businesses plan for network maintenance and growth.

2. **Predict Network Performance of Zones Using Time-Series Data**
   Our goal is to anticipate network performance in different zones over time using machine learning prediction models. Our model will forecast future network performance based on current 5G performance data, offering insights into:
   a. Potential network issues or bottlenecks in specific zones
   b. Anticipated service quality in each area

By proactively detecting any network problems before they affect consumers, this goal enables network operators to increase the overall dependability and consistency of 5G services.

❖ Benefits

Achieving these goals will result in:

- **Enhanced Network Planning:** By analysing and forecasting performance variances, engineers and telecom providers may better plan for network upgrades.
- **Better Service Quality**: Businesses may utilise predictive analytics to fix problems before they affect consumers, which increases customer happiness.
- **Efficient Resource Allocation:** Data insights aid in more effective resource allocation, such as placing equipment in high-demand locations or attending to underserved regions.

Our goal with this project is to use data-driven insights to improve the resilience, dependability, and accessibility of 5G networks in different geographic areas.

2. Dataset

2.1 Data Source

The dataset provided for the project consists of network performance data gathered from network testing using 5G trucks in a variety of geographic areas. Each record contains data regarding different network

parameters which allows for a comprehensive understanding of network metrics associated with geographical zones.

The primary features include:

- Time information: Consists of a ‘time’ column in UNIX format along with datetime attributes ‘Day’, ‘Year’, ‘Month’, ‘Date’, ‘hour’, ‘min’, ‘sec’.
- Location data: GPS coordinates ‘latitude’ and ‘longitude’ to track to the position of each truck when data was recorded.
- Truck information: ‘speed’ to represent the movement of the truck and a ‘truck’ column for unique identifier in each truck.
- Server measurements: Data for four different servers ‘svr1’, ‘svr2’, ‘svr3’, ‘svr4’.
- Data transfer information: ‘Transfer size’, ‘Transfer unit’, ‘Bitrate’, ‘bitrate_unit’ to measure the size and rate of data transfer.
- Retransmissions: ‘Retransmissions’ and ‘CWnd’ to track the flow control and network congestion within the network.
- Receive data: ‘Transfer size-RX', ‘Transfer unit-RX', ‘Bitrate-RX’, ‘bitrate_unit-RX' for incoming and receiving data transfer information.

This dataset provides extensive network feature information to allow for geographical clustering of network performance as well as time-series predictions. Performance patterns can be analyzed for network quality improvement and coverage across different zones.

2.2 Data Processing

The raw dataset was run through a multitude of pre-processing steps to ensure higher data quality and usability for modeling:

- Dropping missing rows and duplicates: All rows with missing data were dropped along with duplicate rows to reduce the redundancy of the dataset.
- Filtering invalid coordinates: For invalid GPS coordinates, the dataset contained latitude and longitude values set to ‘99.999’, these values were filtered out to leave only valid location data.
- Feature engineering: Columns not necessary for modeling or analysis such as ‘time’ and ‘timezone’ were dropped.
- Column renaming: Certain column names were renamed to be compatible with the ‘to_datetime’ function. (‘Date’ to ‘day’, ‘min’ to minute’, ‘sec’ to ‘second’.
- Datetime column created: Date and time column values combined to create ‘datetime’ column which allows for easier time-series modeling.

For time-series forecasting specifically, a custom function was used to apply a sliding window to the data. A window size of 3600 representing 1 hour, the data is separated into 1-hour intervals so that network performance for the upcoming hour can be based on recent history of the past five hours.

3. AI Model Development

3.1 Feature Engineering / Feature Extraction / Image Processing

Several feature engineering steps were taken to standardize and prepare features for modeling and analysis:

- Feature extraction: A set of features were selected to capture network performance and data transfer information about each record. Features include server measurements, ‘Transfer size’, ‘Bitrate’, ‘Transfer size-RX' and ‘Bitrate-RX’.
- Standardization: Values were standardized using sklearn ‘StandardScaler’ function. Features were normalized to provide a standardized dataset that is prepared for clustering.

3.2 Train/Test Split

For time-series forecasting, the dataset was divided into training, validation and test sets to train and then test model performance on unseen data.

A train, validation, test split of 80/10/10 was used to allow the majority of the data to be used for training, and equal portions of data for training validation as well as unseen data testing. Instead of random shuffling, a split by time is required for the specific application of time-series analysis.

3.3 Training Model

The training of two primary models is included, a KMeans clustering model for network performance grouping by geographical zones as well as an LSTM neural network model for time-series forecasting of network metrics.

KMeans Clustering Model

The scikit-learn library was used to implement KMeans and Matplotlib to visualize the elbow method. To tune the appropriate ‘k’ hyperparameter, the elbow method was used by calculating the inertia for clustering in ‘k’ values from 2-14. A visualization of the elbow method is provided to pick an appropriate ‘k’ with a high number of clusters and low inertia.

The KMeans model is then fit to the standardized features and clusters are assigned based on network performance characteristics.

LSTM Time-Series Forecasting Model

TensorFlow and Keras libraries were used for building, training and evaluating the model. The model architecture includes an **input layer** that takes historical network performance values, an **LSTM layer**, and **two dense layers**, one with ReLU activattion as well as an output dense layer with linear activation to predict performance metrics in the next hour.

Number of epochs can be selected by the user, allowing for lower or shorter training based on relative model performance. Learning rate set to 0.0001 using the Adam optimizer to prevent overfitting and a

batch size of 32 was used for lower memory consumption. Loss and accuracy are tracked during training against the validation dataset.

3.4 Evaluation of AI Model

KMeans Clustering

The evaluation metric chosen for this clustering model is Inertia, which measures the sm of squared distances between data points and its centroid. Lower inertia indicates better clustering. The ‘elbow’ point in the elbow method plot shows that the number of clusters selected at an intermediate ‘k’ yields well separated clusters and better interpretability.

LSTM Time-Series Forecasting

The primary evaluation metric for this model was Root Mean Square Error, which is a standard deviation of the prediction errors produced by the model. This metric was used to evaluate the trained model against the test set to represent how well the model could predict future network performance data based on historical information.

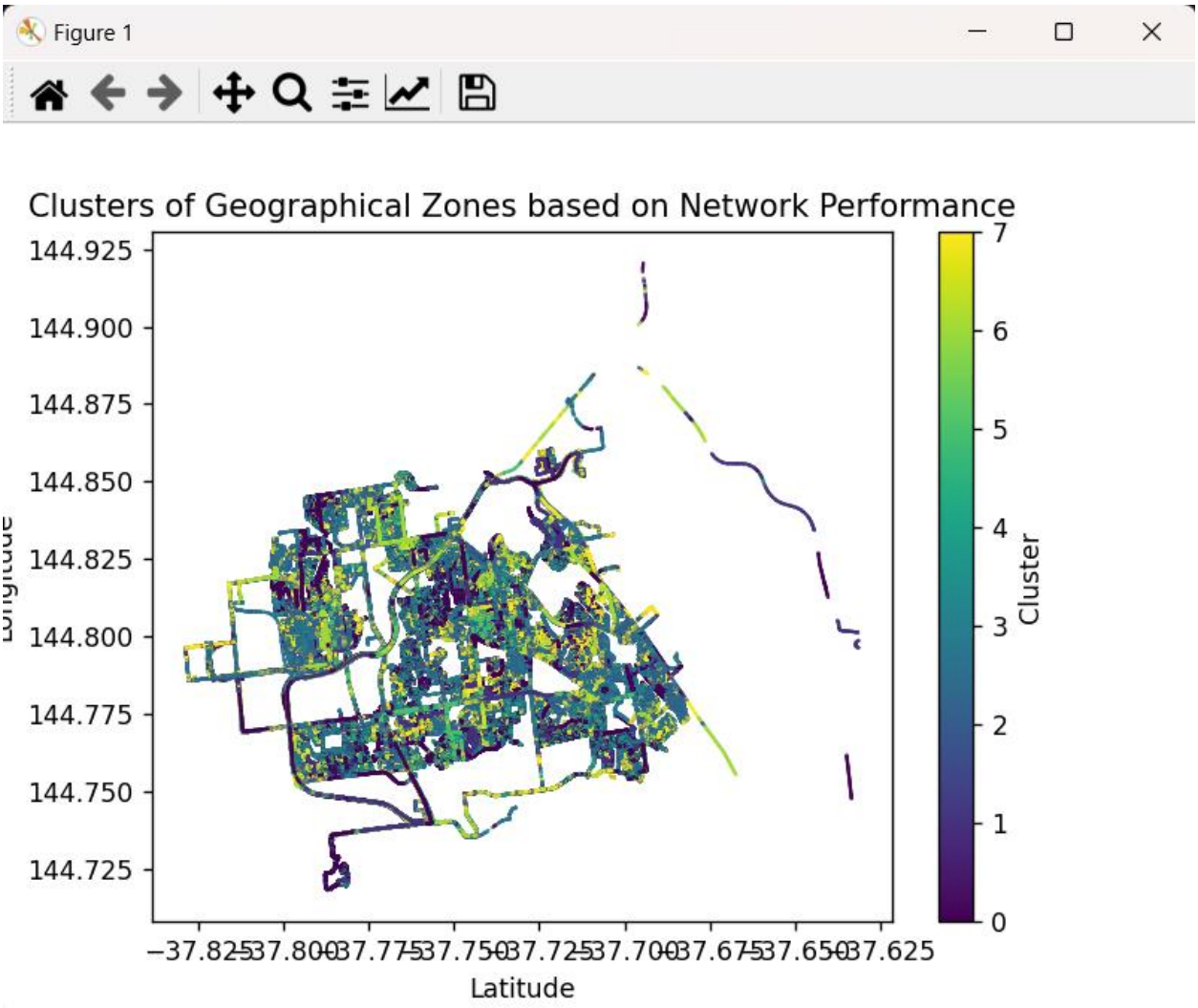
The model’s low RMSE values confirmed that it was able to predict temporal data with relatively low rate of error which leaves it suitable for this forecasting task.

4. AI Demonstrator

Clustering Model Demonstration

The inputs for the demonstrator include latitude and longitude coordinates as well as the standardized network performance metrics for clustering. Using the KMeans clustering model, the network performance data is grouped into geographical clusters and are displayed in a scatter plot to represent specific locations.

The scatter plot visualizes the clustering results with each scatter point representing the color of the cluster that it belongs to, and a color bar indicates the colors by which each cluster can be identified, to allow them to be distinguished easily.



Time-Series Forecasting Demonstration

The inputs for this demonstration include historical network performance metrics as well as a chosen time window, set at 3600 seconds for which predictions are made for the next hour. The LSTM model takes a window of past network performance data and then predicts the network performance for the next hour, the predictions are compared to the previous hour to visualize the results.

A time-series plot is generated to compare the actual previous hour data against the prediction next hour values. The visualization allows users to see how the LSTM model tracks network performance overtime.

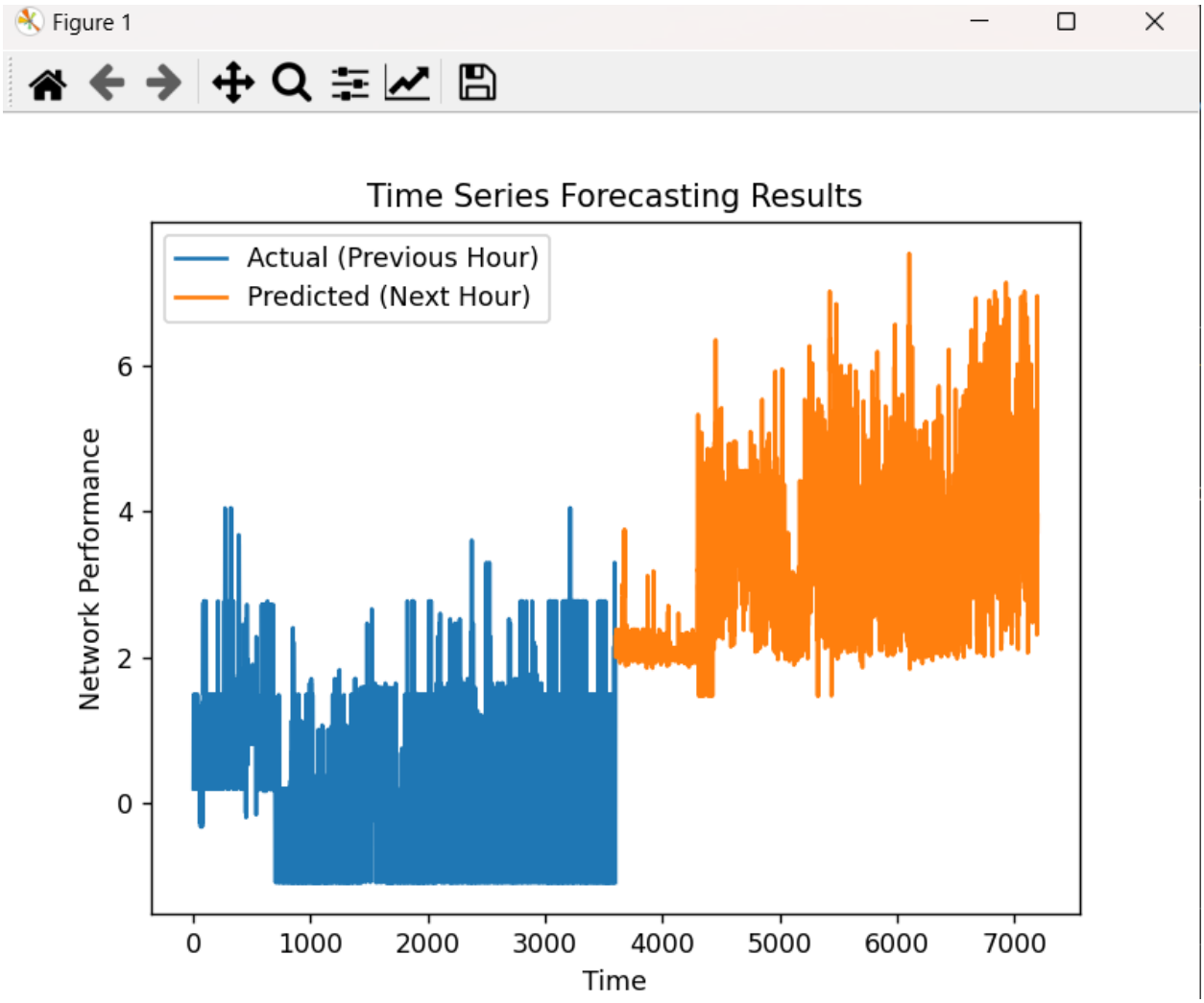


Figure 1
Time Series Forecasting Results
Actual (Previous Hour)
Predicted (Next Hour)
Network Performance
0
2
4
6
0
1000
2000
3000
4000
5000
6000
7000
Time

5. Conclusions

The project managed to achieve the primary goal of clustering geographical zones based on network performance metrics and the development of a time-series forecasting model to predict short term future network performance.

The size of the dataset was quite large; however, data preprocessing reduced the scale of the dataset by a relatively large margin. The speed of the trucks was a detail that was noticed, latency data may be misleading at certain points in time if the truck is mobile within two different network towers and identifying impactful effects requires trial and error.

Clustering analysis revealed insights into geographical zones where network performance metrics were similar, perhaps due to environmental or infrastructure differences. Time-series forecasting indicated that past performance can impact future metrics and appropriate planning can provide for better load management.

Overall, this project has been fruitful in learning about data analysis and machine learning models in predictive as well as geographical analytics.

6. Appendix A

https://drive.google.com/drive/folders/1-1qbYF79TR7LJQXfvBRzz-g18JcBrX-K?usp=sharing

Breakdown of folder:

UI.py - Source code for data processing, training, predictions and visualizations.

All_data.csv - Raw dataset.

LSTM.h5 - LSTM model trained for 10 epochs.

LSTM_100.h5 - LSTM model trained for 100 epochs.

Clustering.png - Screenshot of clustering scatterplot.

Time_Series.png - Screenshot of time-series forecasting prediction.

Read-datasets.py - Source code for combining separate raw data into a single file.