Project

Web and Mobil GIS

Author: Johanna Schaefer

13.09.2025

1 Introduction

2 Methodology

Descriptive analysis

B. Problem formulation

Formulate your speed or flow prediction problem and justify it by discussing its reasonableness for practical use in real-time application. For example,

What to predict? – Do you predict flow, speeds, or both? What is the problem type? – Formulate as a clustering problem, a regression problem, a classification problem, or a combination of these? What features to use? - Develop a-priori hypotheses about features you think are important in predicting 15, 30, 60 minutes in the future.

C. Model development

Data Preprocessing: Clean and preprocess the dataset if necessary, data normalization, and converting categorical variables into suitable formats for modeling. Feature Engineering: Analyze the dataset to identify relevant features that could impact bus arrival times. Create new features if necessary, such as time-based features, distance-related features, and aggregation of historical data. Exploratory Data Analysis (EDA): Perform EDA to gain insights into the relationships between different variables and bus arrival times. Visualize patterns, correlations, and outliers in the data. Model Selection: Select appropriate machine learning algorithms for short-term prediction task. Consider conventional machine learning models or deep neural network models depending on the nature of the data. Model Training: Split the dataset into training and validation sets. Train the selected models using the training data and fine-tune hyperparameters to achieve optimal performance. Model Evaluation: Evaluate the trained models using appropriate evaluation metrics such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE). Compare the performance of different models to select the best-performing one. For a comparative analysis, you need to develop at least three different types of models. D. Model diagnostics

Design different scenarios to explore how the best model works overall. When and where it does not work? Is it robust to noisy and missing data in real-time? Is it generalizable over months?

E. Model deployment

Would you recommend your model to the agency to use? Reflect on what you learned in the project and what you envision as an innovative way to operationalize it in practice.

F. Report

Put your code with comments on your GitHub repository and save your best-trained model to an external file (.keras format, https://keras.io/api/saving/model$_{s}aving_{a}nd_{l}oading/).$

Submit a report that addresses tasks A) to E) and answers all questions, including the following:

Brief introduction and project objectives Descriptive analysis Model formulation Model development Model diagnostics Model deployment Summary of main findings and future works

GitHub code link

3 GitHub code link