

Team: Dhriti Gada

Since I am the only member of my team, all of these tasks are assigned to me.

Data Collection and Preprocessing

1. **Define goals for this project** - Speak with a rep from the metro to figure out what I should focus on for this project.
 2. **Obtain raw bus data** - Retrieve historical bus data (including schedules, locations, and sensor data). I am currently in communication with the head of public communication at the Cincy metro.
 3. **Clean and preprocess bus data** - Perform data cleaning, including handling missing values, removing outliers, and standardizing the dataset for model input.
 4. **Split dataset into training, validation, and test sets** - Ensure that the dataset is split appropriately. Need to figure out whether a validation set is needed for this type of project
-

Model Design and Development

5. **Define deep learning architecture** - Design a suitable neural network architecture (e.g., CNN, RNN, or LSTM) based on the project goals (e.g., predicting bus delays, optimizing routes).
 6. **Select loss function and optimizer** - Choose an appropriate loss function and optimizer based on the model's objectives.
 7. **Develop model implementation in PyTorch/TensorFlow** - Implement the deep learning model in the chosen framework
 8. **Integrate real-time data for model updates** - Set up data pipelines that continuously feed new bus data into the model for periodic updates and training. Must determine if this is possible or if only historical data will be available to me.
-

Model Training and Evaluation

9. **Train deep learning model on training data** - Train the deep learning model using the training dataset
10. **Evaluate model on validation set** - Test the model on the validation set after each epoch and fine-tune hyperparameters to prevent overfitting.
11. **Perform model hyperparameter optimization** - Use techniques such as grid search or random search to find the best-performing hyperparameters for the model.
12. **Monitor training performance using TensorBoard** - Track training progress, visualizations, and key metrics using TensorBoard to ensure the model is learning correctly.

-
- 13. Evaluate model on test set** - Once the model is trained, evaluate its performance on the test set to measure its generalization ability.
-

Model Refinement and Deployment

- 14. Refine model architecture based on performance** - Modify the model's architecture, such as adding layers or adjusting activation functions, to enhance performance based on evaluation results.
- 15. Convert model to deployment-ready format** – Look into model deployment techniques and formats and find one that is suitable for this project
-

Documentation and Reporting

- 16. Generate performance metrics report** - Summarize the model's performance on validation and test sets, including accuracy, precision, recall, and other relevant metrics.
- 17. Write final project report** - Compile a detailed report summarizing the project, including the objectives, methodology, challenges, results, and future directions.
- 18. Prepare deployment and usage documentation** – Create documentation that outlines how users can interact with this model