**COS30018 Intelligent Systems TP 1 2024**

**Title: Assignment 1 Option A Topic 2**

****

**Name (Student ID): Joseph Linao (104556329)**

**Abdur Rehman Shah Hamdani (104170609)**

**Adonias Pedro (104463681)**

**Lab class: Tuesday / 10:30 AM to 12:30 PM / ATC627**

**Due Date: Friday 27th  October 2024 at 11:59 PM**

**Table of Contents**

1. **Introduction**

* **Overview of traffic congestion**
* **Motivation for AI-Driven Traffic Management**
* **Goals and Objectives of the Project**
* **Scope of AI techniques used**
* **Structure of the Report**

1. **Overall System Architecture** 
   * **Systems Components**
   * **Data flow and communication**
   * **Technology stack**
2. **Implemented interactions protocols** 
   * **Agent Communication**
   * **Decision-making Processes**
3. **Implemented search / optimisation techniques** 
   * **Machine Learning Algorithms**
   * **Evolutionary Algorithms**
   * **Graph-Based Search for Route Optimisation (if necessary)**
   * **Performance Evaluation Metrics**
4. **Experimentation** 
   * **Case scenarios**
   * **Simulations**
   * **Results**
5. **Critical analysis of the implementation** 
   * **Strengths**
   * **Limitations**
   * **Comparisons**
6. **Summary** 
   * **Key Points**
   * **Key Learnings**
   * **Team Member Contribution**
7. **Conclusion**
8. **Appendix**

1. **Introduction**

**Overview of traffic congestion**

In 2016, traffic congestion in Australia cost over $19 million, with Infrastructure Australia projecting this could rise to $39.8 billion by 2031 without further infrastructure investment. Congestion is inevitable in large cities, mainly due to supply and demand imbalances in road capacity, along with accidents, road works, and weather. This project, led by intelligent systems students, explores how AI applications can enhance traffic management efficiency to address congestion challenges.

**Motivation**

As traffic demand continues to grow, congestion often arises because roadways and interactions are not designed to accommodate the actual volume of vehicles, particularly during peak hours.

**Goals and objectives**

Goals:

* Develop an AI agent to efficiently reduce traffic congestion in urban areas, focusing on cities in Victoria.
* Improve the affordability, efficiency, and reliability of traffic management by integrating intelligent systems and machine learning models for dynamic signal control and adaptive routing.
* Optimise traffic flow and reduce travel times using real-time data and predictive analysis for better transportation management.

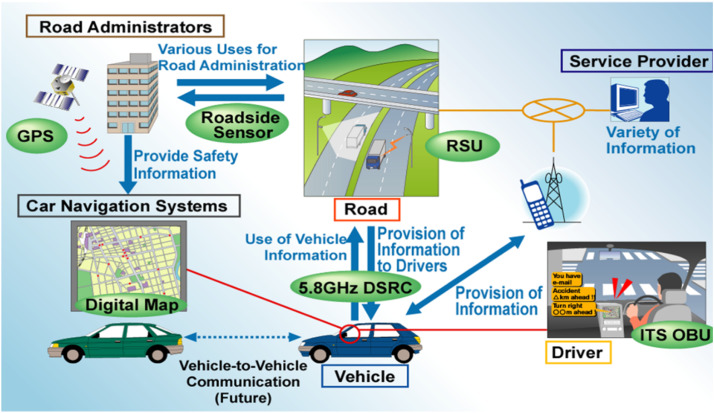
Objectives:

* Develop a basic traffic flow congestion system capable of predicting traffic volume at specific intersections.
* Implement different machine learning (ML) techniques to improve prediction accuracy
* Develop a GUI for user intersection, enabling input configuration and visualisations of results.
* Bonus: Create a visual map based on the coordinates of a dataset

**Scope of AI technique used**

The project uses multi-agent systems, reinforcement learning, self-supervised learning, and evolutionary algorithms to optimise traffic flow and congestion.

**B. Overall system Architecture**



**Systems Components**

The system comprises of:

* Data input: Historical time data from the Scat Data in October
* Preprocessing: Data cleaning, normalising, and formatting for model training
* Model training: ML models like SAE (Sparse auto-encoder), LSTM (Long-Short Term Memory), GRU (Gated recurrent unit) for prediction including a custom model

**Data flow and communication**

The data flows from input (historical datasets such as the Scat Data October) to preprocessing, followed model training, and finally to the user interface for predictions.

**Technology stack**

* Python libraries:
  + Tensorflow
  + Keras
  + Matplotlib
  + Scikit-learn
  + Warnings
  + Extension: Folium
* Dash (Plotly) or Tkinter for GUI

**C. Implemented Interaction Protocols**

**Agent communications**

Agents handle tasks like signal control, route optimisation, and incident management. Communication protocols include:

* Message Passing: Agents exchange structured messages for tasks like adjusting signal testings.
* Standard Passing: Agents uses FIPA/ACL for consistent
* Hierarchical Model: A central agent directs coordinate agents
* Real-time sharing: Agents share traffic data continuously for rapid decision.

Decision Making Processes:

Key Processes to optimise traffic flow include:

* Data analysis: Agents gather and analyse data to detect patterns and congestion.
* Collaborative Response: Agents coordinate for rerouting and signal adjustments.
* Adaptive Signals: Signals timings adjust in real-time based on traffic conditions.

**D. Implemented search / optimisation techniques**



**Machine Learning Techniques**

* Stacked Autoencoder (SAE): excel at feature extraction, enabling the model to pearl high level representations of traffic patterns, leading to more accurate predictions.
* Long Short-Term Memory (LSTM): capable of handling long-term dependencies, which are crucial in traffic flow prediction, as they consider the influence of past traffic congestion on current flow.
* Conventional Neural Network (CNN): provide high accuracy in detecting and classifying objects, which is essential for real-time monitoring in traffic management.
* Reinforcement Learning (RL): is highly adaptable, continuously learning from feedback to improve its decision-making over time.

**Evolutionary Algorithms**

* Genetic Algorithms (GA): well-suited for complex, multi-objective optimisation problems, making them effective in scenarios where traditional methods may struggle
* Particle Swarm Optimisation (PSO): explains a wide solution space efficiently, making it effective for optimising complex systems with many variables.

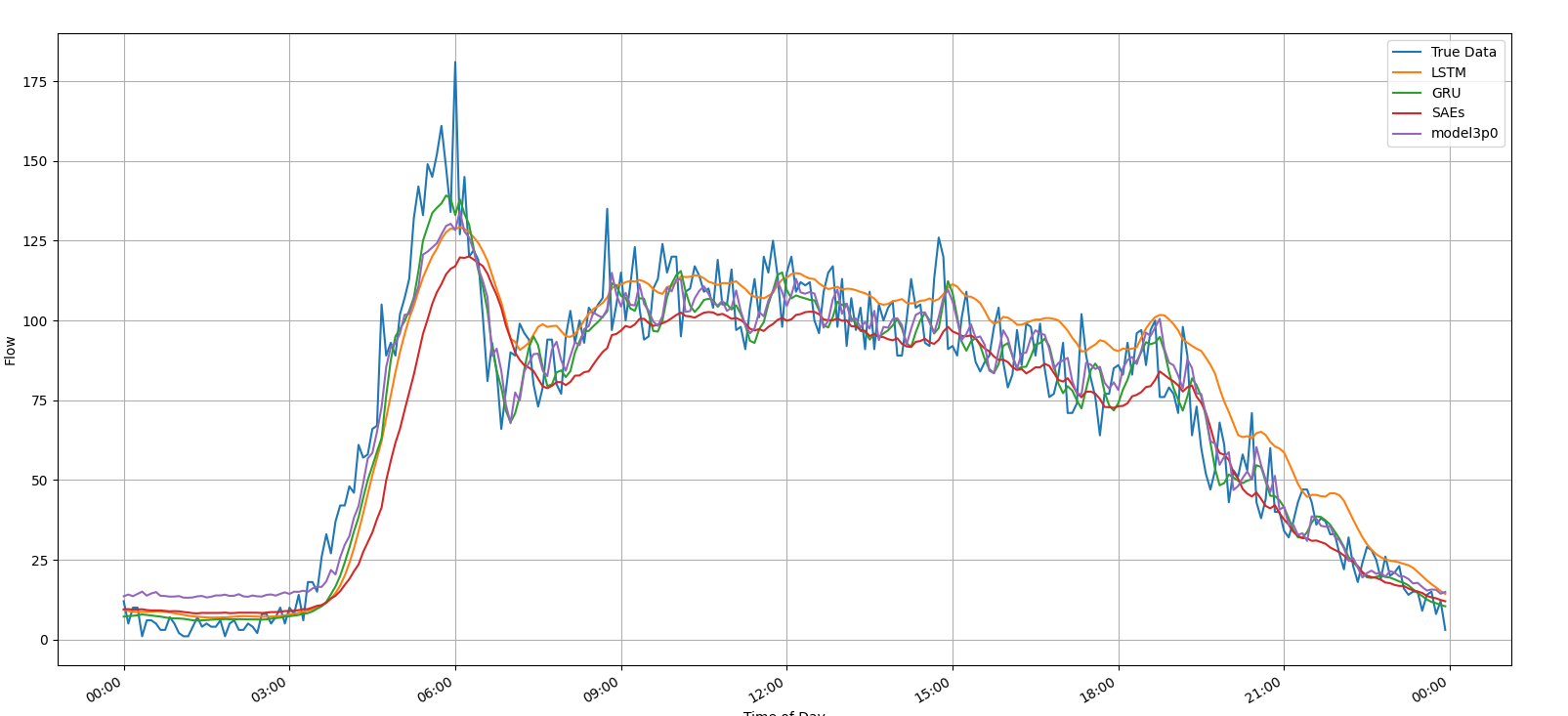
**E. Experimentation**

**Models Configurations and Set-up**

Besides minor re-configuration and set-ups, this is how to set up the traffic flow congestion management:

1. Open Anaconda Prompt (Windows)
2. Create a new environment: conda create <name your created environment> python=3.6
3. Activate the environment: conda activate <name your environment>
4. Install the following packages (keras, tensorflow, scikit-learn)
5. Follow the repository (<https://github.com/xiaochus/TrafficFlowPrediction>) for further instructions
6. Download the git repository inside a folder where it will be easy to find and access e.g. ‘*C:\Users\name\repos\*’
7. Use the train.py file to train each model in the program ‘python train.py –model [model name]’
8. And you can the run main file by using the command ‘python main.py’ and it will the graph for all the models in the program
9. To run the GUI is ‘python gui.py’. Go to the ‘Map Generation’ tab, load data for the map first (scats\_data\_e, scats\_data\_n, scats\_data\_s, scats\_data\_w). Choose a start location and and location and then click on show route to show a map. Click on the blue mark to see the traffic flow prediction.

**Demonstration & Comparison**



* LTSM: best at tracking true patterns, with good accuracy during peak hours but slightly overestimates during late hours
* GRU: follows trends but is more conservative, understanding some peak flows
* SAE’s: Most conservative, smoothing short-term fluctuations but understanding peaks, with the largest deviations.
* Model3p0: Similar to GRU, moderately accurate, better at capturing fluctuations than SAEs.

Extension option 3 uses the \*\*consolidated\_filtered\_dataset.csv\*\*, extracted from the Python file \*\*Extract.py\*\*, based on location and coordinates (latitude and longitude). This dataset offers high-resolution traffic data from multiple sensors, capturing variations over time. Each entry records sensor data at fixed intervals, structured to reflect time steps, sensors, and features, providing a consistent view of traffic dynamics.

**F. Critical analysis**

**Strengths**

* High prediction accuracy: Advanced models like SAEs and LSTM improve accuracy terminal patterns, aiding in complex traffic management.
* Real-Time Adaptability: Reinforcement Learning (RL) adapts to real-time variations, crucial in handling rapid shifts in traffic.
* Robust Optimisation: Evolutionary Algorithms (e.g., GA, PSO) optimise signal timings and routes, ensuring balanced flow and reduced travel time.
* Scalability: Modular design and standardised protocols enable seamless integration and expansion to larger networks.

**Limitations**

* High Computational Demand: Deep learning models like LSTM and SAE require substantial computing power, slowing response times during peak hours
* Static Route Optimisation: Current optimisation assumes static conditions, reducing accuracy in unpredictable and system robustness.

**Opportunities for improvement**

* Comprehensive Datasets: Incorporating diverse data (e.g., weather, accidents) could improve prediction accuracy and system robustness
* Hybrid Models: Combining LSTM with CNNs can better capture spatial-temporal data, enhancing prediction accuracy.
* Enhanced User Interface: Improved UI with features like heat maps and real-time alerts could boost user intersection and system utility.

**G. Summary**

The Traffic Flow Prediction System (TFPS) enhances urban traffic management by using AI for accurate traffic prediction, adaptive signal control, and route optimization.

**Key Findings:**

* Accurate Predictions: SAE and LSTM models provide high prediction accuracy, improving traffic management and reducing congestion.
* Real-Time Adaptability: RL ensures dynamic response to traffic changes, enhancing efficiency
* Effective Optimization: Evolutionary Algorithms optimises routes and signals, ensuring balanced traffic flows.
* Scalability: Modular Design supports easy integration of new models, agents, or datasets.

**Key Learnings:**

* Model Complexity vs Performance: Balancing model accuracy with real-time performance is critical
* Importance of Diverse Datasets: Broader Datasets (e.g., weather, road conditions) improve accuracy and generalisation.
* Continuous Learning: RL emphasises the need for ongoing adaptation to enhance decision-making

**Team member contribution:**

| **Joseph Linao (104556329)** | * Project report:   + Introduction   + Overall System architecture   + Implementation search / optimisation techniques   + Summary   + Experimentation * Data preprocessing * Extract - Location based on directions (N, S, E, W) * Main file - using virtual environment * Creation of the custom model * Extension 3: Map * ⅓ of the GUI - Created a function where the destination from point A to B is visually presented. |
| --- | --- |
| **Adonias Pedro (104463681)** | * Project report:   + Experimentation   + Critical analysis   + Summary   + Conclusion * Data preprocessing * Extracted scats and locations from the scats data october 2006 excel file * Modify the GUI * Train and Tested each csv file with an accuracy * Evaluating the map & Outputting the Flow Prediction values |
| **Abdur Rehman Shah Hamdani (104170609)** | * Project report:   + Experimentation   + Summary   + Appendix * Responsible for the GUI * Training the models * Adjusting the models * Loading the data into the GUI |

**H. Conclusion**

The AI-driven Traffic Flow Prediction System effectively demonstrates how intelligent systems can optimise traffic flow, reduce congestion, and improve urban mobility. By using machine learning models like LSTM, GRU, and SAE, it provides accurate predictions and route optimization. Each model's performance is analysed, highlighting LSTM's peak capability, GRU's trend accuracy, and SAE's stability. Though successful, there is room for improvement in handling peak and late-hour estimations. The project showcases AI's potential in intelligent traffic management, laying a foundation for future research in sustainable transportation.

**I. Appendix**

Journals and Conference Papers:

"Effective Measures for Traffic Congestion Management" - IJETT Journal. <https://ijettjournal.org/Volume-71/Issue-3/IJETT-V71I3P244.pdf>

Akhilesh, R., et al. (2018). "Urban Traffic Flow Prediction using Machine Learning Techniques." ACM Digital Library. <https://dl.acm.org/doi/10.1145/3219819.3220096>

Lambert, J., et al. (2015). "Managing Traffic Congestion: Strategies and Techniques." Transportation Research Part A: Policy and Practice. <https://onlinepubs.trb.org/Onlinepubs/trr/1992/1365/1365-011.pdf>

Hall, R., et al. (1992). "Estimating Travel Times in Urban Environments." Transportation Research Record. Available online. <https://onlinepubs.trb.org/Onlinepubs/trr/1992/1365/1365-011.pdf>

Wang, Y., et al. (2019). "Real-Time Traffic Management with IoT and AI Solutions." IEEE Xplore. View document. <https://ieeexplore.ieee.org/document/8669845>

*Traffic Congestion Relief Associated with Public Transport: State-of-the-Art and Future Directions* by Nguyen-Phuoc et al. (2020). This paper reviews the impact of public transport on alleviating traffic congestion and suggests future research avenues.<https://link.springer.com/article/10.1007/s12469-020-00231-3>

Web resources:

*Traffic Congestion Overview and Insights* - ScienceDirect. <https://www.sciencedirect.com/topics/social-sciences/traffic-congestion>

*Traffic Congestion Solutions in Australia* - iMOVE Australia. <https://imoveaustralia.com/topics/traffic-congestion/>

Australasian Transport Research Forum (2022). "Innovations in Urban Traffic Management." <https://australasiantransportresearchforum.org.au/wp-content/uploads/2022/05/ATRF2022_Resubmission_85.pdf>

*How to Fix Congestion – Transportation Policy Research* by Texas A&M Transportation Institute. This report outlines different policy strategies to address traffic congestion.

<https://policy.tti.tamu.edu/congestion/how-to-fix-congestion/>