Milestone-2 - Literature Review - Individual Report

From the literature review, I gathered that the paper [Elmi21] introduces TRES-Net, a groundbreaking deep learning architecture tailored for taxi fare prediction, which effectively integrates Residual Networks (ResNet) and Bi-directional Long Short-Term Memory (Bi-LSTM). TRES-Net uniquely focuses on managing spatial-temporal dependencies, creating a matrix of similar trips based on the road network structure, and utilizing a periodically shifted attention mechanism to handle long-term periodic dependencies. Additionally, it incorporates external factors like holidays and events, capturing similarities across various temporal dimensions. The comprehensive evaluation demonstrated TRES-Net's superiority over existing models, showcasing its impressive performance metrics, including Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). Furthermore, the paper highlights the potential for real-time adaptation, suggesting avenues for future research.

The paper [Bagal23], on the other hand, delves into the application of Deep Neural Networks (DNNs) for taxi fare prediction, outlining the implementation details of the "Parallel-DNN" algorithm. It emphasizes the adaptability of DNNs in handling diverse data types, showcasing their ability to manage numerical data efficiently. The study effectively demonstrates the advantages of DNNs over other conventional algorithms, displaying superior accuracy and robustness, particularly in managing large datasets. The paper's focus on the practical implications of leveraging DNNs for enhancing service quality and pricing strategies within the taxi industry further solidifies the significance of this research.

I also contributed to the implementation part. Regarding our implementation, we are going to strategically amalgamate various technologies and methodologies highlighted in the literature. Leveraging resources like NYC-TLC S3, AWS EMR, Apache Spark, AWS Sagemaker, Pandas, Python, and Flask, we aim to develop a well-optimized system. Our approach prioritizes efficient batch processing, effective data preprocessing techniques, and streamlined data storage and retrieval processes. By employing memory-based processing techniques and reducing excessive disk-based I/O, we expect significant improvements in overall system performance and speed. Additionally, our user-centric approach, with a Python Flask web application integrated with the Google Maps API, aims to ensure a user-friendly interface, thereby enhancing the practical applicability and user experience of our system.