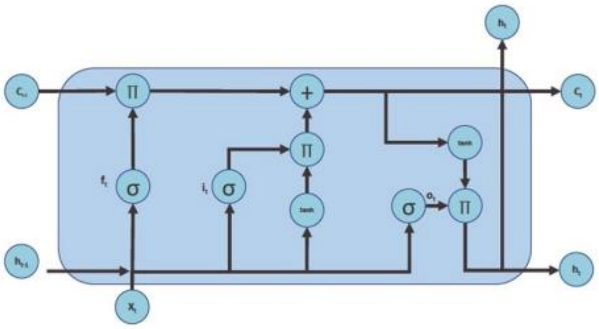
POLITECNICO
MILANO 1863

Artificial Intelligence in Decision Making and Control of Industrial Applications [2023-24]

Assignment 3: Sequence Modelling (LSTM)



Task Overview:
The objective of this task is sequence modelling using a Long Short-Term Memory (LSTM) network to predict the movement of a vehicle. Specifically, the goal is to accurately forecast the car's coordinates for the next few (i.e., at least 5) time steps.

Dataset: vehicle_data.zip

Dataset Description:
The dataset consists of 18,457 files. Each file contains data for 12 data (i.e., labels), observed over 67 time steps. It includes:

- Coordinates (x, y) of the car.
- Velocity.
- Acceleration.
- Distance from the centerline.
- Indicators if there's a car on the left, right, or front.
- Distance from your vehicle to other cars.

Requirements:

- Split the original dataset into three training, validation, and test sets to correctly train and evaluate models.
- Design and implement LSTM model/s to be trained for predicting the (x, y) coordinates of the vehicle for the next few (i.e., at least 5) time steps.
- Evaluate the performance of the model/s using suitable error metrics like the Root Mean Square Error (RMSE) or any other loss function you find appropriate.
- You may also use other sequence modelling architectures/models (e.g., GRU) and compare it with the LSTM-based ones.

- Report your work in this section, including pseudocodes, codes, results, and points/conclusions, in a Jupyter Notebook or a PDF/HTML.

Consider:

- You may use any programming language or tool. For instance, you may use Python or MATLAB; There are various packages and libraries available for both platforms for implementing your models, including TensorFlow and PyTorch for Python or MATLAB's Deep Learning Toolbox.
- While the dataset provides a range of data (i.e., labels), you can decide how many of them you want to use. You might choose to use only the car's coordinates, or you could incorporate more information to potentially enhance your model's predictions.
- Although the data is continuous across many files, it does restart at random points. It is recommended to consider each file (12 labels × 67 observations) as an individual training unit. From this, use the first 12 labels × 62 observations as the input and the corresponding 2 labels × 5 observations as the response. Once you have your model's predicted 2 labels × 5 observations, compare it to the original to calculate the prediction error.
- You may employ various techniques (e.g., data preprocessing) and methods to enhance your models and the task. Accordingly, you can create a benchmark.
- You can utilize Google Colab, which offers free CPU/GPU resources for Python using Jupyter Notebook.
- You are required not to use LLMs (e.g., ChatGPT) in any capacity for solving this assignment.
- You should submit your assignment on Webeep.

Deadline:

- Sunday, February 18th, 2024. *(Originally announced on Jan 23rd!)*
- Disadvantage (10%): Sunday, February 25th, 2024.