The year 2020 was a catastrophic year for humanity. Pneumonia of unknown aetiology was first reported in December 2019., since then, COVID-19 spread to the whole world and became a global pandemic. More than 200 countries were affected due to pandemic and many countries were trying to save precious lives of their people by imposing travel restrictions, quarantines, social distances, event postponements and lockdowns to prevent the spread of the virus. However, due to lackadaisical attitude, efforts attempted by the governments were jeopardised, thus, predisposing to the wide spread of virus and lost of lives.

The scientists believed that the absence of AI assisted automated tracking and predicting system is the cause of the wide spread of COVID-19 pandemic. Hence, the scientist proposed the usage of deep learning model to predict the daily COVID cases to determine if travel bans should be imposed or rescinded

**Thus, your task is to create a deep learning model using LSTM neural network to predict new cases (*cases_new*) in Malaysia using the past 30 days of number of cases. You only need these 4 columns as the features: *(cases_new, cases_import, cases_recovered, cases_active)*.**

**There are TWO types of scenarios you need to consider:**

1. **Single step window – input width of 30, output width of 30, offset of 1**
2. **Multi-step window – input width of 30, output width of 30, offset of 30**

Following are the criteria of the models.

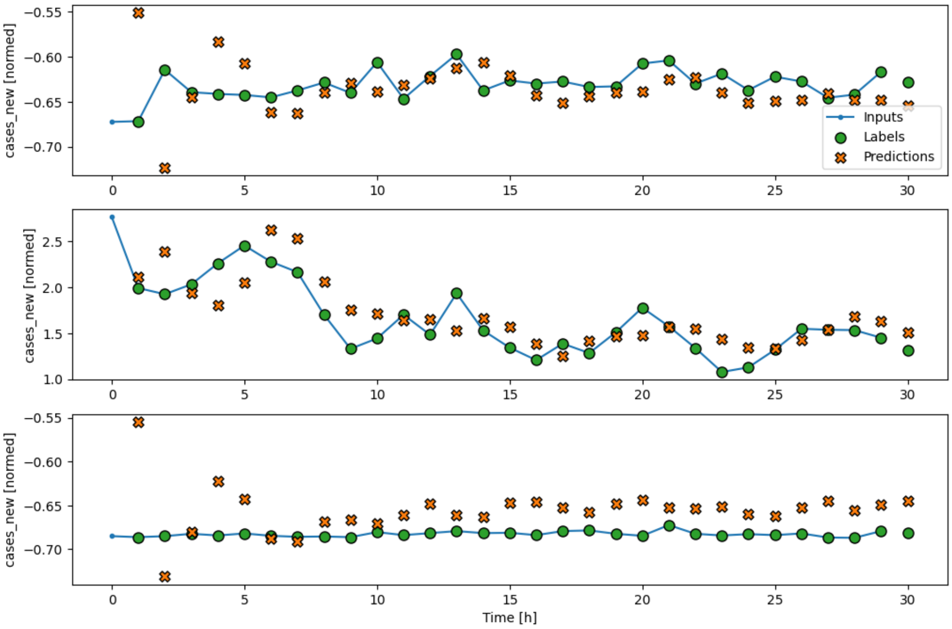
1) There will be 2 models, one for each window scenario.
2) Only LSTM, Dense, and Dropout layers should be implemented in the model.
3) The depth of the model can be set according to your needs.
4) Window size configuration should follow the instructions above.
5) MAPE error should be lesser than 1% when tested against testing dataset.

$$Mean\ Absolute\ Percentage\ Error = \frac{Mean\ Absolute\ Error}{sum(abs(y\_actual))} * 100\%$$

6) Training loss should be displayed using TensorBoard. ***Note: Be careful with this! Make sure you setup your TensorBoard callback object separately for both models so you can display them individually.**

Files to be submitted and uploaded to GitHub and LMS (submission link will be given on the assessment day):

1) Main python file. (GitHub and LMS)
2) Classes of the python file (GitHub and LMS)
3) Dataset (GitHub and LMS)
4) Architecture of the models saved as .png should be included in README.md and be displayed on your GitHub. (GitHub and LMS)
5) Training process plotted using Tensorboard can be snipped and saved as image file format (LMS).
6) Performance of the models and the reports can be snipped and saved as image file to be included in the zip folder for LMS submission. (LMS and GitHub)
7) Include your GitHub URL directing to your assessment 4 in a text file then submit to LMS. (LMS)
8) **Don't forget to credit/cite the source of the data on your GitHub page** GitHub - MoH-Malaysia/covid19-public: Official data on the COVID-19 epidemic in Malaysia. Powered by CPRC, CPRC Hospital System, MKAK, and MySejahtera.
9) A graph showing the predicted and actual covid cases included in the results section in README.md. The graph shown below is an example: (GitHub and LMS)



***Note: You need the graph to display test data! You can achieve so by modifying the *example* property in your *WindowGenerator()* class.**

*Please zip all the required files into one folder then submit to LMS.

**Please save the dataset and model in 2 different folders to GitHub.

Complete the assessment and submit the files to LMS and GitHub by 5pm. Good Luck!!!

| | 100% | 50% | 0% |
|---|---|---|---|
| Task Completion (30%) | Scripts can be executed without any error on trainer’s local machine. | - | Scripts fail to be executed on trainer’s local machine. |
| Project requirements (30%) | Able to achieve the objectives of the project using relevant and appropriate approach. | Able to achieve the objectives of the project but using inappropriate approach such as brute forcing the solution. | Fail to achieve the objectives of the project. |
| Exploratory data analysis (30%) | Demonstrates strong understanding on the objectives of the project and performs relevant approach to process the data. Necessary data processing techniques such as, data loading, data cleaning, features selection and data preprocessing are performed and well justified. | Shows comprehensive understanding of the objectives of the project but uses incorrect or irrelevant approach to process the data. For example, removing NaN data when there are limited amount of samples in the dataset. | Shows limited understanding of the objectives of the project. Absence of data processing section in the code. |
| Code readability (5%) | Involves the usage of functions or methods for repeated tasks. Codes are easily readable and justified by including comments and description texts. | Minimal usage of functions or methods for repeated tasks. Available comments and descriptions but lack of details. | No usage of functions or methods for repeated tasks. Codes are difficult to read and understand. Missing descriptions and comments. |
| GitHub repo (4%) | Detailed and clear instructions of the project on README.md. Results such as graphs are also included in README.md as part of the project description. | Project successfully uploaded to GitHub repo but with incomplete README.md. Missing descriptions, instructions, and results. | Fails to upload project to GitHub repo and missing README.md |
| PEP8 compliance (1%) | Fully complies with PEP 8 Standard | Partially complies with PEP 8 Standard | Fails to comply with PEP 8 Standard |
| Total (100%) | | | |