MACHINE LEARNING I PROJECT

Subject 1: Hanoi Temperature Forecasting

Let’s build some nice application to predict Hanoi temperature.

Step 1: You need to collect Hanoi temperature and raw weather features from the following website.

The weather data that you can find here is either daily or hourly. Let’s start with daily data first and collect 10 year data of Hanoi weather data. For weather forecasting, especially in such climate change, you need a very long data to understand the movement.

https://www.visualcrossing.com/weather-query-builder/Hanoi/us/last15days/

Step 2: Data Understanding: The dataset has 33 columns (features). Explain the meaning and values in each column. For example, what does feature “moonphase” mean? What are its values? Try to plot the target column (the Hanoi temperature). What is your observation about Hanoi temperature for the last 10 years? Try to understand the relationship between different features of weather (their correlation maybe). How they can combine together to detect Hanoi temperature.

Step 3: Data Processing: You have to process your data, determine feature type (numerical, categorical), handle missing values, compute correlation matrix, so on and so forth to normalize and understand more your data.

Step 4: Feature Engineering: Here comes the interesting part of the job. First, you have to ask yourself what forecasting/predicting means in this particular context. Fore example, a good explanation of forecasting here is to use daily data and try to figure out the Hanoi temperature for the next 5 days. Note that, the output of Feature Engineering module is to transform your raw data into a dataset of features that you are available to throw into a Machine Learning model to get a certain prediction.

- In this dataset you will have some text features as well. How to leverage those features to predict Hanoi temperature.

Step 5: Model training and hyper-parameter tuning:

- Well, this part, feel free to use any kind of ML models which get you the best prediction. Try to use Cloud-based framework like ClearML to monitor your ML model.
- You can use Optuna as a good framework for hyper-parameter tuning.
- Talking about tuning/optimization, we talk about various metrics to evaluate your model (RMSE, MAPE, R2, etc). Try to use them all, understand and interprete them in this particular context.
- How to split data into train, test, val set providing that you do not have some sort of data leakage problem.

Step 6: Build a UI to demo your app (Streamlit, Gradio, Reflexe, Chainlit, etc).

Step 7: Let’s dive deeper into training system. We care about your model performance over time. The common sense is, if you train model and use it to predict day by day, at some point, the performance will downgrade. When you should retrain your model?

Step 8: In past questions, we only mentioned daily data, right. Now, with hourly data in hand, do you think you can do better with Hanoi temperature forecasting. Rerun the whole process with hourly data to see (probably with somewhat new ideas with hourly data's different nature).

Step 9: Study ONNX, figure out when and why use ONNX for your deployment efficiency. Apply to this kind of situation.

Subject 2: Exactly like subject 1, but this time, you consider Sài Gòn temperature.

Note:

Deadline is 23h 09th November 2025

Try to work as a team so that you can learn the subject (ML product) as a whole together, do not just do some question split for each member in the team.

Send the folder result "class_name_group_number" to projects - Google Drive Project code source in .py files. For results, demo and model performance, make some nice, concise report and record a demo video (you can simply send a demo link if you already put your code in cloud).

DSEB 65A:

Subject 1: Group 1, 3, 5, 7

Subject 2: group 2, 4, 6, 8

DSEB 65B:

Subject 1: Group 1, 2, 3, 4

Subject 2: Group 5, 6, 7, 8