Assignment 2 - Part 3

ECSE 552 - Winter 2025

Due date: April 10, at 23:59

Things to submit:

Please submit your Homework in a zip file to myCourses. The zip file name should be `Group_X.zip`, where `X` is your group number. Failure to submit in the required format will incur a deduction of points.

Your `.zip` should include:

1. All source code (`e.g.: ECSE552_W25_Assignment2_part3_Group_X.ipynb`) with saved outputs.
2. A PDF file with your written report (`ECSE552_W25_Assignment2_part3_Group_X_report.pdf`), instructions below. Please type your solutions and upload it as a PDF. You may chose your editor of choice (Word, LaTeX, etc..

You need to submit your code as a proof but **only your PDF report will be graded**. *NOTE:* Do not include code or references to your code (e.g. "see lines 100-120 of `model.py`") in your report.).

Report submission details

Please include both a description and an explanation of your choice for the following requirements:

Method (20 pts)

1. Features
2. Preprocessing steps (if applicable)
3. Specific model architecture (of the 3 models)
4. Loss function(s)
5. Your model (advantages, special properties, difference from the other two, etc.)
6. Hyperparameters and how you selected them
7. Training procedure

Results (20 pts)

Describe and explain the following (15 pts):

1. Metrics used for comparison
2. Model comparisons in terms of the 4 specified attributes (separately)
3. Overall performance comparison

Additionally, describe and explain (5 pts):

1. Error propagation (i.e. attempting to predict values at time t to $t+k$ when you are only given data from time steps $t-k$ to $t-1$). This means that you will have to use the "predicted values" at time t as input[1] to your model to predict the attributes at time $t+1$.

Task

Weather Forecasting using Deep Learning

In this homework, you will attempt to create a time-series forecasting model. Specifically, you will be using the weather time-series dataset [2] **(Use WS Saaleaue)** recorded by the Max Planck Institute for Biogeochemistry from 2009 to 2016. This dataset was prepared by François Chollet and originally has 10-minute intervals. We are only going to do hourly predictions so we removed some of the rows in the dataset attached to this homework.

Given a history of weather attributes from time $t-k$ to $t-1$, you have to predict[3] the following values for time t:

- `p(mbar)`, atmospheric pressure
- `T (degC)`, air temperature
- `rh (%)`, relative humidity
- `wv(m/s)`, wind velocity.

Your input features are the weather attributes of the previous k time steps. You are required to use `p(mbar)`, `T (degC)`, `rh (%)`, and `wv(m/s)` as features. You may include other attributes provided that these are values from the previous time steps.

To reduce search space, you may select a fixed values of $k \in [4, 8]$. You may also preprocess[4] the attributes in any way you want as long as there is no data leakage. For example, you may use the hour in the `Date Time` column since the time of day can affect the temperature. You may also predict other attributes if you think this is useful in terms of learning. However, we will only look the the 4 attributes above for model comparisons.

[1] If your model only predicts the 4 attributes but uses other features, assume that the other features are given
[2] download the data
[3] multi-task learning
[4] hint: time and angles have different notions of similarity compared to temperature

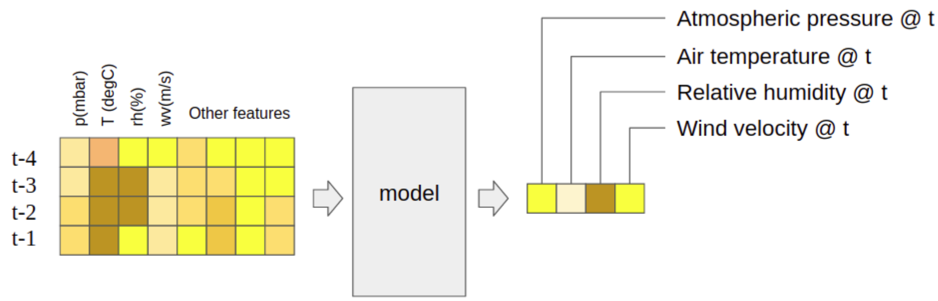

Figure 1: Problem overview with $k = 4$

An illustration of the task is shown in Figure 1.

You have to implement 3 models:

- A baseline model consisting of only fully-connected (`nn.Linear`) layers
- A baseline model that uses at least one layer of LSTM
- Your own model (may use any neural network, including but not limited to CNNs and GRUs)

For the LSTM model, features must be fed to the network sequentially (i.e. from $t - k$ to $t - 1$).

The first 2 models should have a justifiable “default” loss function (e.g. MSE, Cross Entropy, KL Divergence). Your own model may use custom loss functions. There is no limitation on the number of layers, nodes, and regularizations that you are allowed to use provided that they are justified in your report.

To ensure fairness of comparison, the first 2 models should have the same number of layers as your own model. You do not need to perform an extensive hyperparameter search as long as they are reasonable (e.g. fixed batch size, learning rate, etc.). Note that **we do not expect you to create a state-of-the-art model**.

FAQs

- Can we use libraries other than PyTorch?
  Yes.
- Can we use resources other than Colab (*e.g.*, own GPU, Compute Canada)?
  Yes.
- Does the hyperparameter tuning have to be rigorous (e.g., 5-fold nested CV)?
  No. However, we will deduct points for bad ML practices such as having data leakages.
- Can we use the raw version of this data from the original source?
  Yes, but it is not encouraged unless you are going to base your “other analysis” on this raw data. If you are going to use the raw data for performance improvement, you may not use the data from less than 10-minutes of the timepoint to be predicted.
- Can we use other datasets?
  Yes, if they are relevant to the task or to the analysis.

- What is k in the error propagation analysis?
  It is up to you. We purposefully made it vague because some of you might want to expand this analysis.
- What is the format of the report?
  It is up to you. As long as the required details are there.
- Should we use teacher forcing?
  Up to you.
- Can we use AI tools to write the report?
  We have no intention of checking whether the report was AI-generated. We will treat your report as something you have written. However, we will check if your code corresponds to your report.