Bidirectional LSTM for time-series forecasting

A case study on Ozone data of Ticino, Switzerland

Anubhab Biswas

anubhab.biswas@usi.ch
Faculty of Informatics
Università della Svizzera Italiana(USI)

January 27, 2025

1. Introduction
2. Data background
3. The Model
4. Results

Background and Motivation

- Forecasting is an essential part of time series analysis having widespread applications in finance, weather, hydrology, traffic control, energy sector, health services, transportation, production etc.
- One of the major challenges however lies in the fact that time series data posses a unique feature - they are serially related. This means that the most basic assumption of independence of the data is no longer valid.
- When building a deep learning model to be used for forecasting this has to be kept in mind as previous values of the target variable $y_{t-L}, \ldots, y_t$ will act as features for forecasting $\{y_{t+1}, \ldots, y_{t+h}\}$.
- A bunch of deep learning models have been developed for forecasting in the past few years such as RNNs, LSTMs, GNNs, GRUs, TCNs, Wavelet networks, Transformers, DeepAR, etc.

Why Bidirectional LSTM?

- Bidirection LSTM has the advantage of capturing dependencies in the data both using past and future instances, which makes it understand relationship from both sides.
- Being still an LSTM it can model sequential patterns, for example, seasonal trend or irregular fluctuations better than a CNN or RNN.
- The bidirectional nature can be useful when the underlying process is non-stationary and non-linear since it can adapt better to changes in the data shift and data characteristics.

Main Objective

- To accurately forecast the 24-hour ahead ozone levels at the different monitoring stations around Ticino.

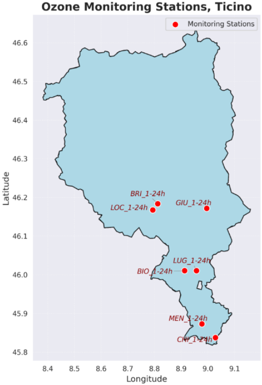

Figure 1: OASI Stations for Ozone monitoring

Practical Constraints

- In the study the ozone data from the monitoring stations of Airolo, Avengo-Cimetta, Bodily, Chimney, Noise, Magadino, Moesano hill, Mount Generoso, Pregassona, Roveredo and San Vittore were not conisdered.
- The research is focused on the southern part of Ticino, centering around the Luganese-Mendrisiotto belt - which is densely-populated and has a high traffic; making it more important for administrative purposes than the mountainous and sparsely populated northern region.
- Due to its photochemical origin, ozone shows a strong seasonal pattern, with higher concentrations in summer. For this reason we focused our analysis only on the period from May to September in the years between 2015 and 2021.
- Although we do not use the ozone data from the northern Ticino, a lot of covariates and measurements related to the weather from the weather monitoring stations of north Ticino are indeed considered as features in our study.

Features

- A large number of features for example nitrogen oxide (NO), nitrogen dioxide (NO_2), global irradiance (G), atmospheric pressure (P), relative humidity (RH), temperature (T), dew point (TD), cloud cover (CN), wind speed (W_s) etc.
- This resulted in the high dimensionality of the dataset, and we had to reduce the overall number of features.
- For minimizing the computational effort we partly replaced the hourly values of the measured and forecasted signals with basic statistical aggregations, i.e. minimum, maximum, and average values over a longer time period.
- Based on suggestions from experts in the field of atmospheric physics, we further manipulated some of the signals available in the dataset to create additional features.

Feature Selection - Shapley Values Approach

- To reduce the size of the dataset as well as to only consider the features that are important in some way to the variable of interest, viz. ozone level.
- We take a Shapley value approach to evaluate and assign the feature importance - an idea that originated in the field of game theory where they are used to estimate the contribution of various agents in increasing the welfare of a community.
- $$\phi_i(f, x_{\text{tr}}) = \sum_{z \subseteq x} \frac{|z|!\,(k - |z| - 1)!}{k!} \left[f(z, \Theta) - f(z \setminus i, \Theta) \right]$$

  where $f(x_{tr}, \Theta)$ is a regression model, which in our case is the NGBoost algorithm.
- k : no. of variables in the training set x_{tr}, $z \setminus i$: subtraction of the i^{th} feature from the reduced dataset z.
- Package used : *shap*

Data Imputation

- There were some significant inconsistencies in the dataset in terms of missing values and we imputed them by using a model-base Random forest Regressor imputation.
- **Rolling Median Calculation**: Compute 24-hour rolling medians for features and add them as new columns.
- **Feature Engineering**: Add time-based features like hour, day of the week, and month to enrich the data.
- **Model-Based Imputation**: Train a RandomForestRegressor on non-missing data and predict missing values using the remaining features.
- **Data Update**: Replace missing values with predicted values, ensuring completeness and consistency.
- Package used: *RandomForestRegressor* from *sklearn.ensemble*

BiLSTM Layer

- Processes main time series data bidirectionally for richer temporal dependencies.
- Outputs hidden states for both forward and backward passes.
- Is followed by a fully connected layer which outputs mean and log-variance predictions for each forecasting step.

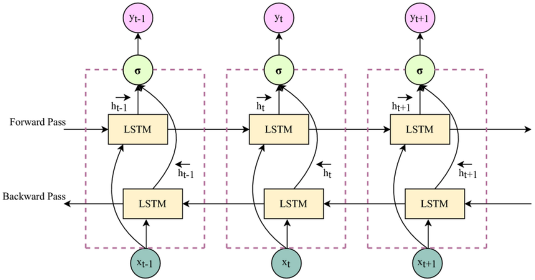

Figure 2: BiLSTM Architecture

Creating linear dependencies

- The post-processing FLAP requires a lot of linear combinations of the original signals in order to produce accurate forecasts.
- These linear combinations are created in two ways - principal components analysis (PCA) and simulation of linear coefficients from Gaussian distribution.
  - **Principal Components Analysis (PCA):** The linear coefficients (principal components) are extracted by decomposing the covariance matrix of the original signals, capturing the key patterns as weighted combinations of the original features.
  - **Gaussian Coefficients Simulation:** Generates random linear combinations by sampling coefficients from a Gaussian distribution.

FLAP

- FLAP (Forecast Linear Augmented Projection) is a post-processing variance reduction technique.
- The FLAP method reduces forecast variance by adjusting the forecasts of multivariate time series to be consistent with the forecasts of linear combinations (components) of the series generated from the PCA and Gaussian Coefficient simulation using projections.
- We train the model for these linear combinations in the same way as the original signals and obtain the forecast of these linear combinations.
- Using the forecasts from both the original signals and their linear combinations, the FLAP produces a new set of forecasts for the original signals which are consistent with the linear constraints of these linear combinations.

Results

- COMING SOON...

Colab Notebook

- Click on the following colab notebook for a brief demonstration of BiLSTM on the Ozone data of Lugano.

Thank you for listening !

Anubhab Biswas

anubhab.biswas@usi.ch