User manual

To run the application:

1. install conda.

2. Clone this repository.

```
$ git clone https://github.com/stephanpesch/Innolab_08_KI_EE.git
$ cd Innolab_08_KI_EE
```

3. Create an environment, change <env> to whatever name you want to give your environment:

```
$ conda create --name <env> --file requirements.txt
$ conda activate <env>
```

4. To add a OpenWeatherMap Token create a file named "tokens.py" and create the variable "open_weather_map_token" like this:
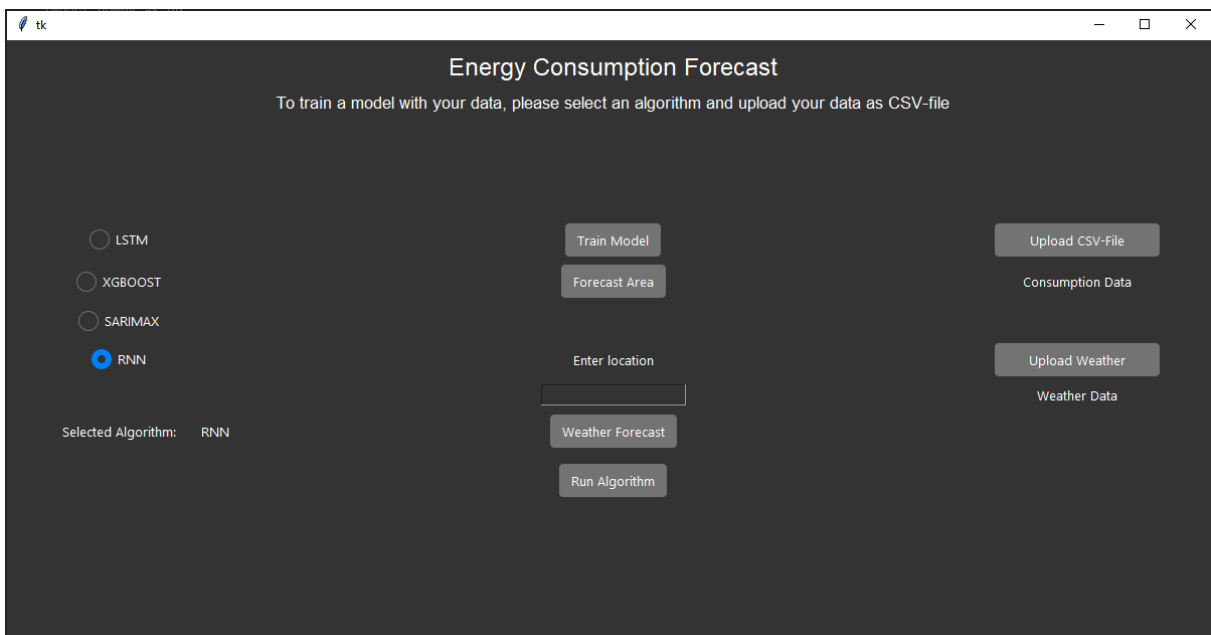
```
open_weather_map_token = "token here"
```

5. Run the application:

```
$ python main.py
```

Starting page

On the starting page, there are multiple options to select from:

1. Train a model
2. Make a prediction
    a. You need to have a trained model before you can make a prediction
3. Get the weather forecast

Train a model

To train a model upload a dataset with the consumption data and a data set with the weather data on the right-hand side of the application.
After that an algorithm should be selected of one of the four algorithms on the left.

Then the "Train model" Button should be clicked.

A screen opens where the features of the dataset should be selected, on the left side you should select the time and the target value you want to have a prediction of, on the right side the time and temperature features that could influence the prediction.

Disclaimer: Ensure that the indices of both CSV files match. It is the user's responsibility to verify and confirm the alignment of the indices.

If you want to perform a grid search, tick the checkbox on the bottom of the screen. If the grid search is finished, the algorithm is executed with the hyperparameters of the grid search.
If you don't want to run a grid search, the algorithm is executed with hardcoded hyperparameters.

Disclaimer: Grid Search can take a long time. Depending on your computer specs and the complexity of the algorithm it can take a few minutes up to <u>several hours</u>.


2 Parameter Selection

Prior to running the algorithm, basic data cleaning is conducted. However, it is essential for the user to independently validate that the dataset being used does not contain any erroneous, corrupted, improperly formatted, duplicated, or incomplete data.

The training returns a model for the forecast area and shows the result of a forecast for a test period.

Make a prediction

If a model has been trained before, the prediction can be started from the start screen (Forecast Area), but before pressing the button, the location for the weather forecast must be entered in the text field below.

The prediction is shown for a period of 40 hours.
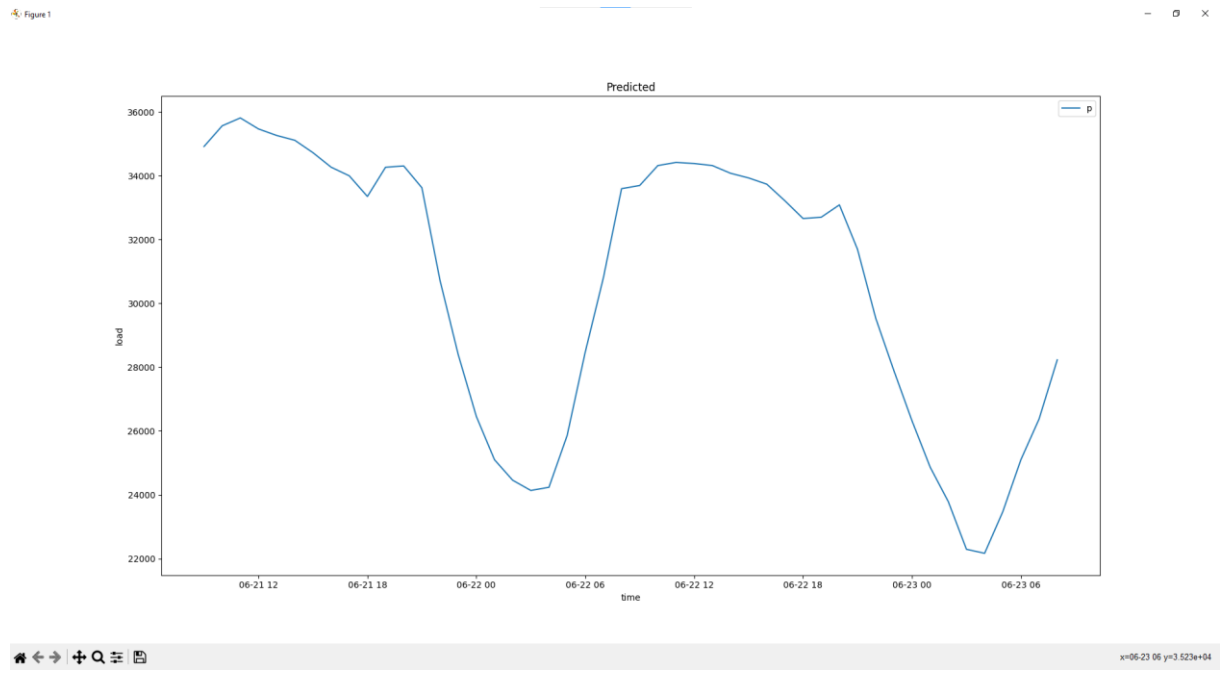
Weather forecast

If a location is entered and the weather forecast button is clicked, the weather forecast for 48 hours is shown.

XGBOOST Constraints

XGBoost makes solid predictions and works with numerous features and even with different targets. A combination or even all these following features can be used: temp, pressure, humidity, wind-speed. The target value can be chosen arbitrarily, but the algorithm works best with the total-load-actual.

SARIMAX Constraints

Sarimax only works with temperature data ("temp"), not with other weather features.

RNN Constraints

RNN can be used to train the dataset with multiple weather features but is not able to give accurate predictions about the future energy consumption. Although it is useful to give information about the future trend. When using the algorithm to predict future energy consumption with the help of the API (Forecast Area) the model must be trained if the temperature data ("temp") only. Using any other combination of weather features will throw an error or non-accurate results.