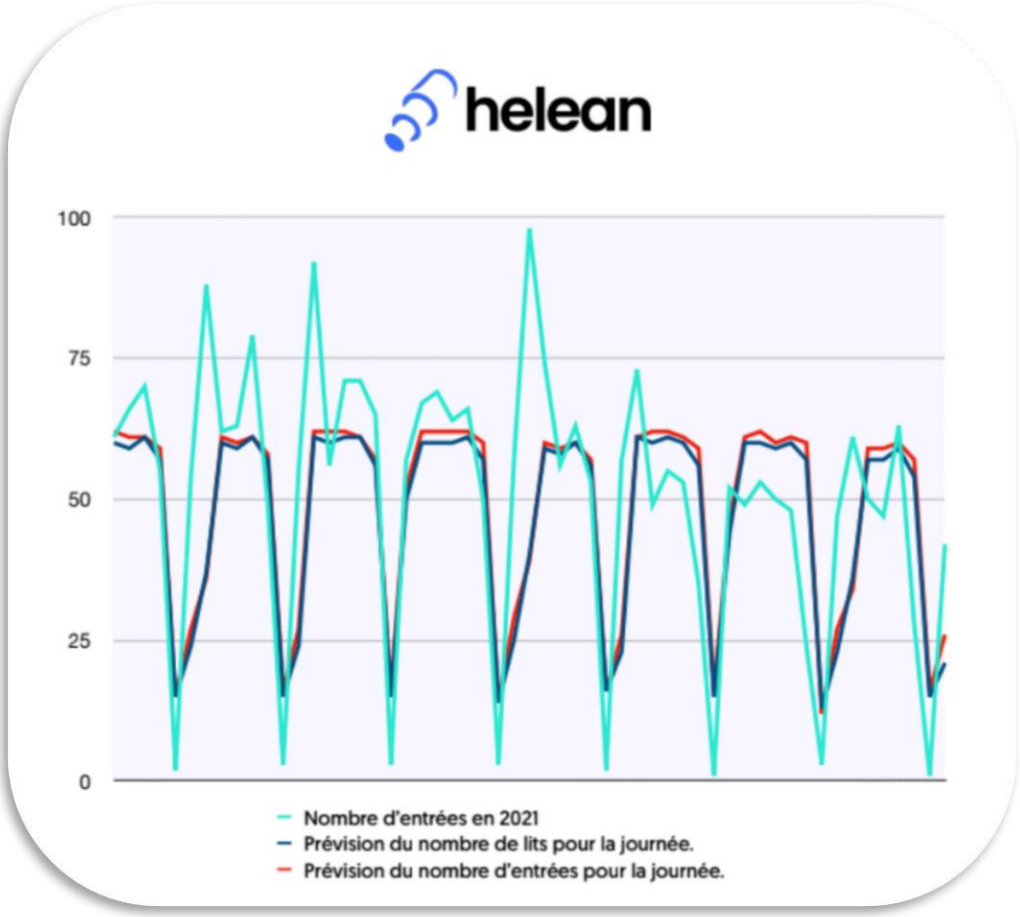


Time prediction forecasting

TEMPORAL PREDICTION OF ENTRIES IN EACH MEDICAL UNIT OF AN HOSPITAL.

[GitHub link]: https://github.com/k0raty/Hospital_prediction_project

Antony Davi | LSTM neural networks | 20 septembre 2022

I.Overview of the program

The maintenance costs of a hospital bed cost between 3000 and 5000 euros per day.

The possibility of predicting the number of beds per medical unit can then prove to be extremely useful. The objective of the project is therefore using data from patients entering over a period ranging from t-n to t, we can predict from t+1 to t+n.

The data are of kind:

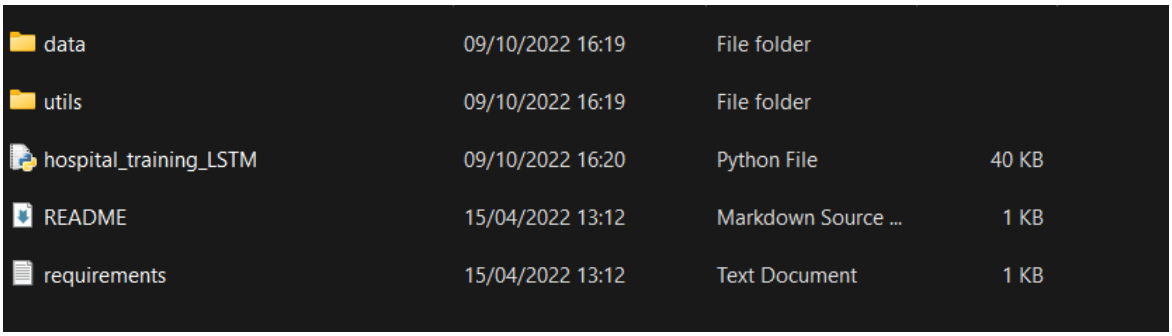
	Num_Index	Finess	Version_Classif	CMD	Sexe	Type_GHM	Complexite_GHM	Num_GHM
0	0	1	11	7	2	C	4	9
1	1	1	11	12	1	C	4	11
2	2	1	11	16	2	M	4	9

[Not every columns are displayed]

1. Each column corresponds to the data of a single patient, data spans 5 years.
2. It refers to the different units visited during his stay, his age, the severity of his complications, etc.
3. Based on this information, considering the calendar, the covid and other exogenous information, we must be able for each unit, to predict the influx for a given time range.

The information relating to the partner hospital and the data used are protected by a confidentiality agreement.

II. Architecture of the program



Hospital_training_LSTM is the main program, it makes it possible to generate data that can be received by the network, trains it but also displays the requested predictions.

This project revolves only around object-oriented programming, the mother classes being HospitalPredictor.

Utils is a folder containing additional programs with useful functions (display, rearrangement of dataframes, etc.)

At initialization, the parent class implements the following attributes in particular:

```
def __init__(self):
    self.data = HospitalDataHandler()
    self.data.build_df() #Build the dataframe with all the UM
    self.params = {
        'T': 20, #Number of days to forcast
        'test_split': 0.15,
        'validation_split': 0.15,
        'verbose': 1,
        'epoch': 10,
        'UM_tot':15

    }
```

HospitalDataHandler is a child class implemented in the data folder in partnership with Helean which carefully coded performs a preprocessing result allowing for each UM and each day to count the number of entrants, their average age, and other averages:

date	gender_avg	severity	age	nb_actions	start_date	end_date	entry_date	exit_date	nb_lits	nb_entry	UM
2019-01-02	0.566666667	0.833333	0.203879	0.237037	0	0	0	0.612946	30	30	0042
2019-01-03	0.483870968	0.860215	0.22956	0.261649	0	0	0	0.672391	31	31	0042
2019-01-04	0.571428571	0.952381	0.168139	0.259259	0	0	0	0.714663	21	21	0042

We decide to train the Vanilla LSTM model [2]:

```
# define model
model = Sequential()
model.add(LSTM(100, activation='relu', return_sequences=True,
model.add(LSTM(100, activation='relu')) #reccurent_activation
model.add(Dense(in_advance))
model.compile(optimizer='adam', loss='mse',metrics=[
    tf.metrics.MeanAbsolutePercentageError(),
    tf.metrics.MeanAbsoluteError()
])
```

This choice results from several failures, it turned out to be the most effective in comparison to Darnn [3] despite the disadvantage that it requires a model by number of days in advance predicted unlike its counterpart.

After tuning hyperparameters, the model can give you a prediction given the following arguments in plot_prediction() function:

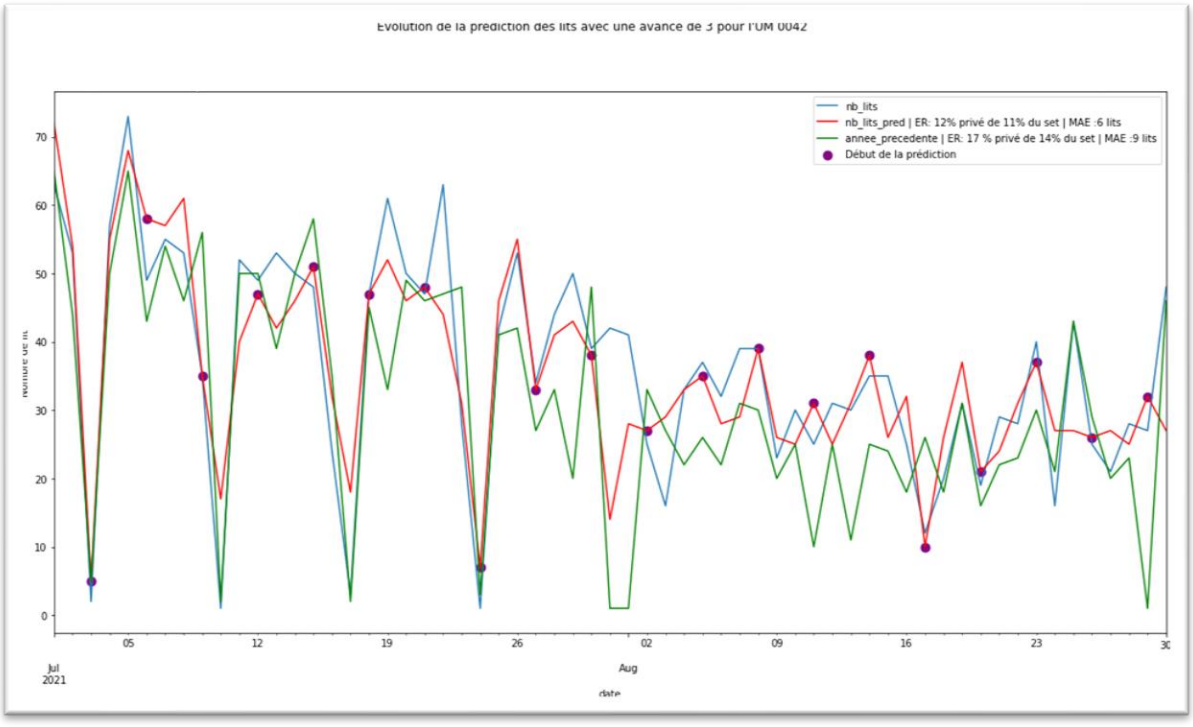
```python
def plot_prediction(self,UM,in_advance,month=None,day=None,from_date:str=None,to_date:str=None,test:bool=False,take_all:bool=False):
    """
    Plot the prediction of a requested UM  and compare it to reality.
    The tuple( from_date , to_date) need to be in the database.

    If the dataframe of prediction wasn't build for the query requested , it creates it.
    -test : Whether , those data have to be taken into account by the model during training
    -UM : A string of the UM (ex : '0036')
    -in_advance : A int corresponding to the range of forcasting
    -month , day are string which will display the months and days requested
    -from_date and to_date are str shaped like : '  2018-11-14 '... , it will display the projection between this dates
    -take_all : Do we have to consider all UM's to train the model ?
    Return a plot

    """
```

The main arguments are UM (which is medical Unit) , in_advance to precise the number of days to forecast in advance knowing the data before. From_date and to_date precise the range of prediction.

Obviously , the model have never been trained on data that he has to forecast.

III.Results :

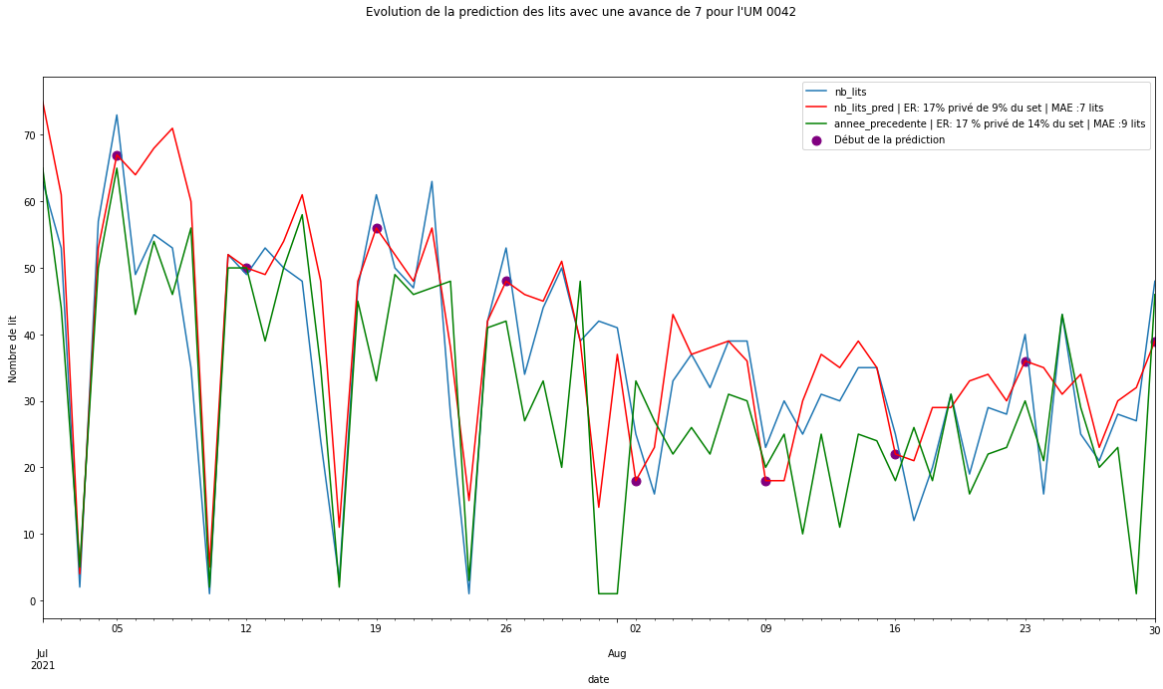
Prediction in august for emergency unit with 3 days of advance



Curve	Relative error	Mean Absolut Error
Previous year (current used model)	17%	9 beds
Red curve (Model prediction)	12 %	6 beds

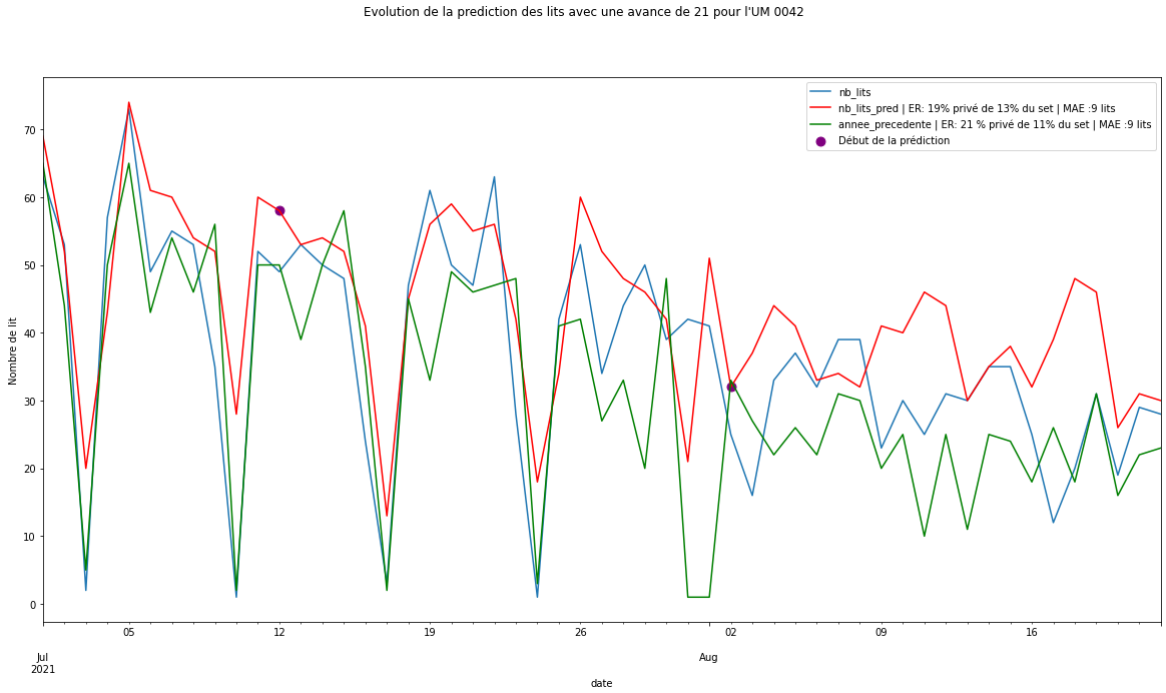
The purple points are "checkpoints", we consider that 20 days of data are known before this day and we forecast 3 days after it (in this case).

Prediction in august for emergency unit with a week of advance



Curve	Relative error	Mean Absolut Error
Previous year (current used model)	17%	9 beds
Red curve (Model prediction)	<17 %	7 beds

Prediction in august for emergency unit with 3 weeks of advance



Curve	Relative error	Mean Absolut Error
Previous year (current used model)	17%	9 beds
Red curve (Model prediction)	20%	9 beds

IV:Conclusion

- The model performs better when it predicts up to 2 weeks ahead.
- It proves its worth when it analyses medical units maintaining a relative periodicity.
- Unfortunately, two weeks is not enough knowing that hospitals are currently only able to provide data every month.

Sources :

[1] : https://fr.quora.com/Combien-co%C3%BBte-un-lit-de-r%C3%A9animation

[2] : https://machinelearningmastery.com/how-to-develop-lstm-models-for-time-series-forecasting/

[3] : https://arxiv.org/pdf/1704.02971.pdf