DTU Compute
Department of Applied Mathematics and Computer Science

Traffic4cast – Traffic Map Forecasting

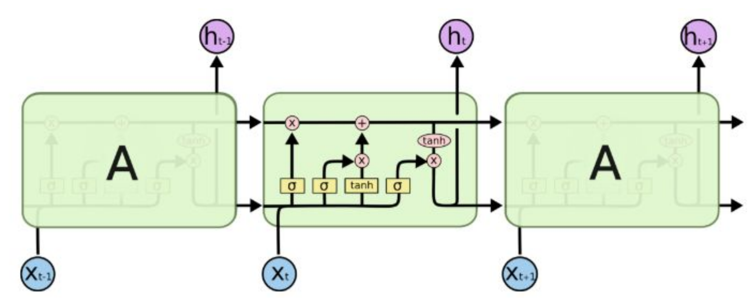
Álvaro Sierra, Arthur Collette, Henrique D.O. Duarte, Ryan Leavitt
DTU Compute, Technical University of Denmark

Introduction

Predicting the next minutes of the traffic in one city can be a challenging task only based on statistical data. However, recurring to sequential-based models it is possible to achieve accurate forecasting results for the next minutes period.
We choose an approach used in the paper: "*Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting*" that captures spatiotemporal correlations better and the other sequential based model or state-of-art algorithms for meteorological prediction.
In this project, we predict the volume of traffic for the next 15 minutes, based on the last hour, for a specific city recurring to **ConvLSTM** cells and the similar model described in mentioned paper.

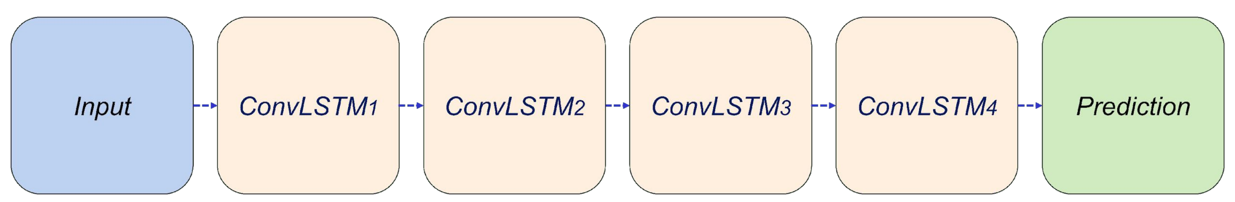
Model

ConvLSTM Cell



The ConvLSTM cell offers convolutional structures in the input-to-state and state-to-state transitions. The LSTM part has the ability to access, write and clear the memory in the cell, previous parameterized in its own gates, and acts as an accumulator of the state information. Every new input passes threw an evaluation, if the accumulator is activated the input it's processed. On another hand, the cell can be cleared if the forgotten gate is on.
One advantage of this approach is that the gradient will be trapped in the cell.

ConvLSTM Layers



The ConvLSTM is used for our spatiotemporal sequence prediction problem. The initial layer propagates the output throw the cells until reaches the final layer that we want to obtain, that is a tensor with our prediction.

Computation

Inputs

We have each city divided in the following list of h5 files:
- Training – 181 days
- Validation – 18 days
- Testing – 163 days

Every training h5 file contains a (288,495,436,9) tensor, and we can describe the encoded data of the last 9 channels:
- [1; 2] - Aggregated volume and speed (i.e. NE)
- [3; 4] - Aggregated volume and speed (i.e. SE)
- [5; 6] - Aggregated volume and speed (i.e. SW)
- [7; 8] - Aggregated volume and speed (i.e. NW)
- [9] - Recording of incidents

Output

We receive a testing file with a tensor of (m, 12, 495, 436, 9), which means that we have to make m different predictions using the last hour (12 x 5).
The final output is a tensor that describes the map of the city, for the next 15 minutes, for each given hour.
In our project, we decide to reduce the size of the map, in order to have better performance and to not consume tones of RAM. In this way, our prediction and training data has a 50x50 size pixel map.

Technical Approach

We opted to use a Google Collab as Runtime Enviroment and the Google Drive to store and load the data needed for each part of the network.
We mostly use a Keras (Tensorflow) library to construct our fully network. The ConvLSTM2D class to construct the layers of our model, as suggests the example of our first layer:

```
convlstm_0 = layers.ConvLSTM2D(filters=32, kernel_size=(7, 7),
      padding='same', return_sequences=True,
      return_state=False, activation='tanh',
      recurrent_activation='hard_sigmoid',
      kernel_initializer='glorot_uniform',
unit_forget_bias=True,
data_format='channels_first', name='convlstm_0')
```

For each prediction, we create 3 images based on the output of the network, i.e. 3*m images. Each image correspond to a 5 time frame.

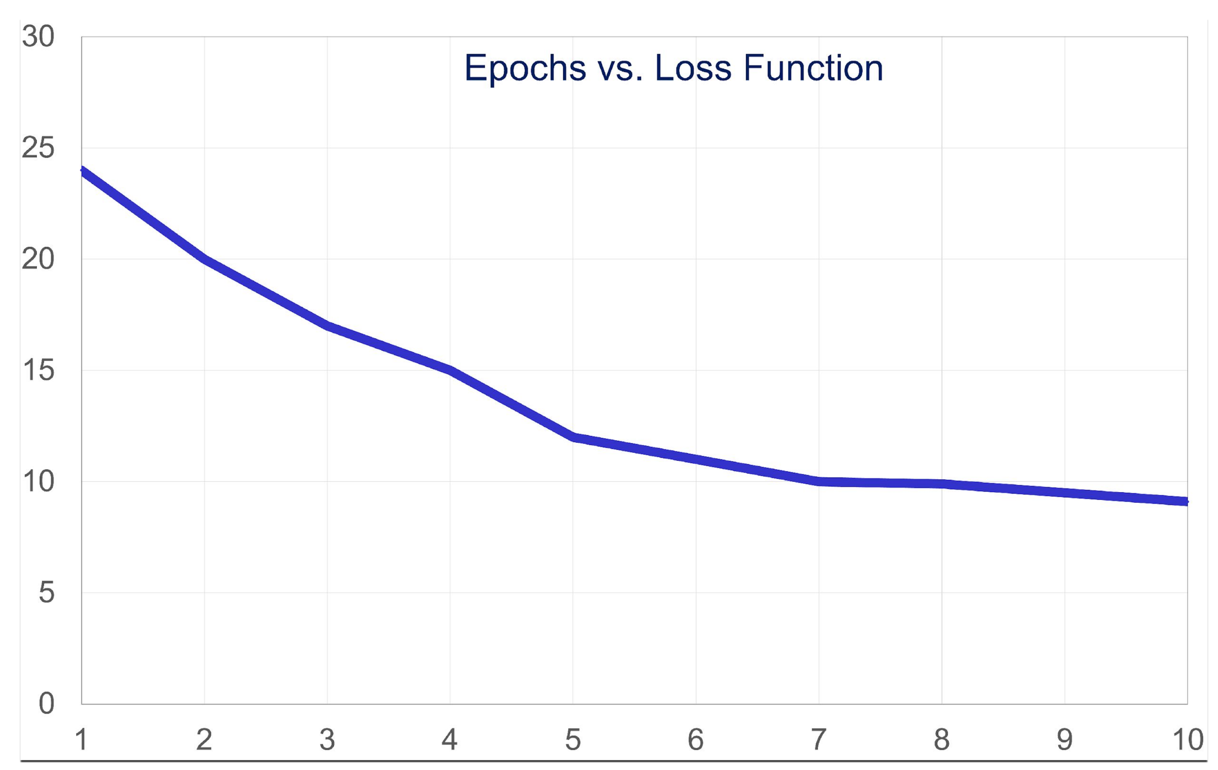
Results

Qualitative

In our qualitative results, we present the 3 next frame images that correspond to 15 minutes predictions.
We can predict the different 8 channels of different aggregated volume or speed for a certain time.
In order to generate images from the data, we need to choose what is the channel that we want to present. In this case, we present the predicted data of the channel 4, which means that the 3 results above, describes the aggregated speed in the SW direction for the next 15 minutes.

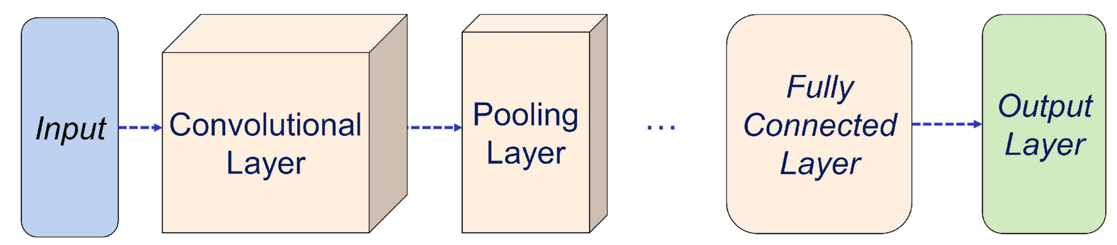
Quantitative

To measure the quality of training the network, we decided to analyse the loss function using the mean squared error function.
We trained our model with a maximum of 10 epochs because after that the loss functions tend to stabilize.



Related Models

Convolutional Network



From our investigation before starting the project, we find that a convolutional network is very popular in achieving results from a spatial tensor or multiple images. However, if we used this type of network to our model, we are losing the temporal properties very relevant to our specific case.

References

[1] Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung. Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting. 2015
[2] Traffic4Cast – Traffic Map Movie Forecasting, 2020 - https://www.iarai.ac.at/traffic4cast/
[3] Keras ConvLSTM2D layer, 2020 - https://keras.io/api/layers/recurrent_layers/conv_lstm2d/