Machine Learning for Industrial Data

Laboratory task № 1

Students:

Kirill Mukhin J4234c

Alexander Petrov J4234c

Alexander Semiletov J4232c

Task formulation. Annotation

- The study of social media streaming data allows us to observe the processes taking place in the city, to predict emerging anomalies and events, allowing us to respond to them in a timely manner.
- Learn how to use historical social network data on the frequency of publications in different urban areas to predict its future distribution.

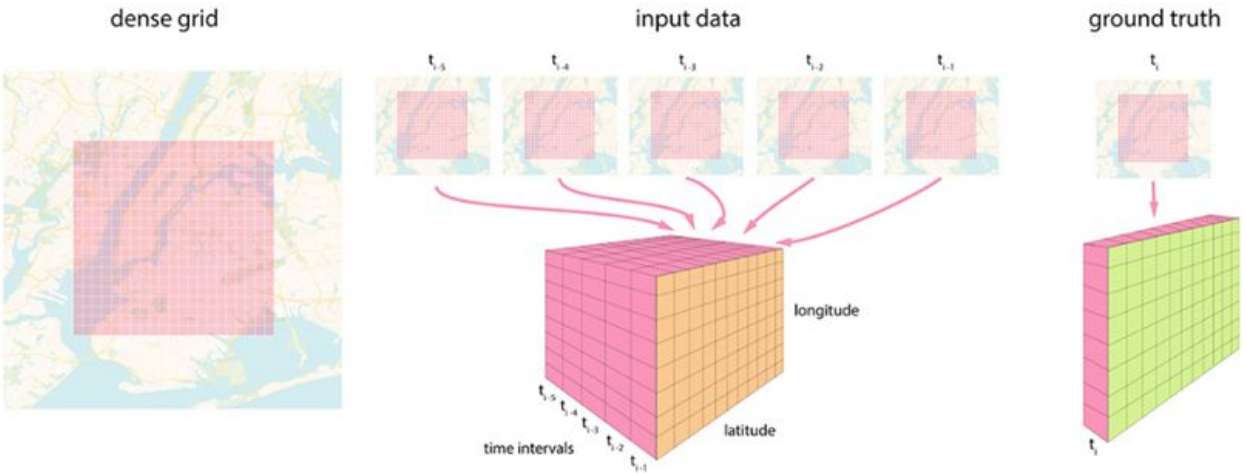


Figure 1 - Data structure observation

Task formulation. Dataset description

- We are presented with a dataset of one popular social network with over 8.5 million meta-information records of publications for 13 months (January 2019 to February 2020).
- Each publication is described with the following metainformation:
  - *lat & lon* - geoposition coordinates rounded to a polygon of 250x250 meters;
  - *timestamp* - time stamp of the publication to the nearest hour;
  - *likescount, commentscount* - publication likes and comments number respectively;
  - *symbols_cnt, words_cnt* - publication symbols and words total number respectively;
  - *hashtag_cnt, mentions_cnt* - hashtags and publication mentioning number;
  - *links_cnt, emoji_cnt* - links and emoji in publication total number;
  - *point* - service field for comparing coordinates from training, validation and test datasets

Data Exploration

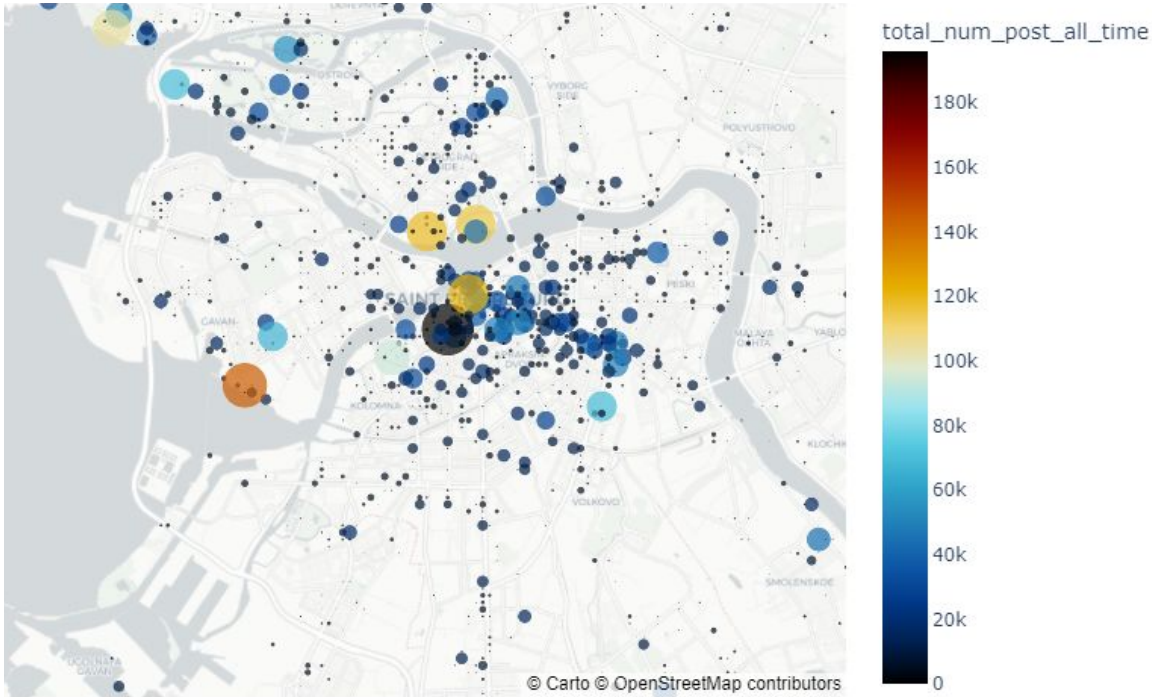


Figure 2 - Display location on the map and number of publications during the entire period

- After filtering data by latitude and longitude based on notes for current lab we have dropped 3 % of train data
- Number of unique points in train dataset is 6658
- Number of unique points in validation dataset is 151
- According to the assignment we have to predict the 13th month's number of publications for each point.

Train & Validation Datasets Unification

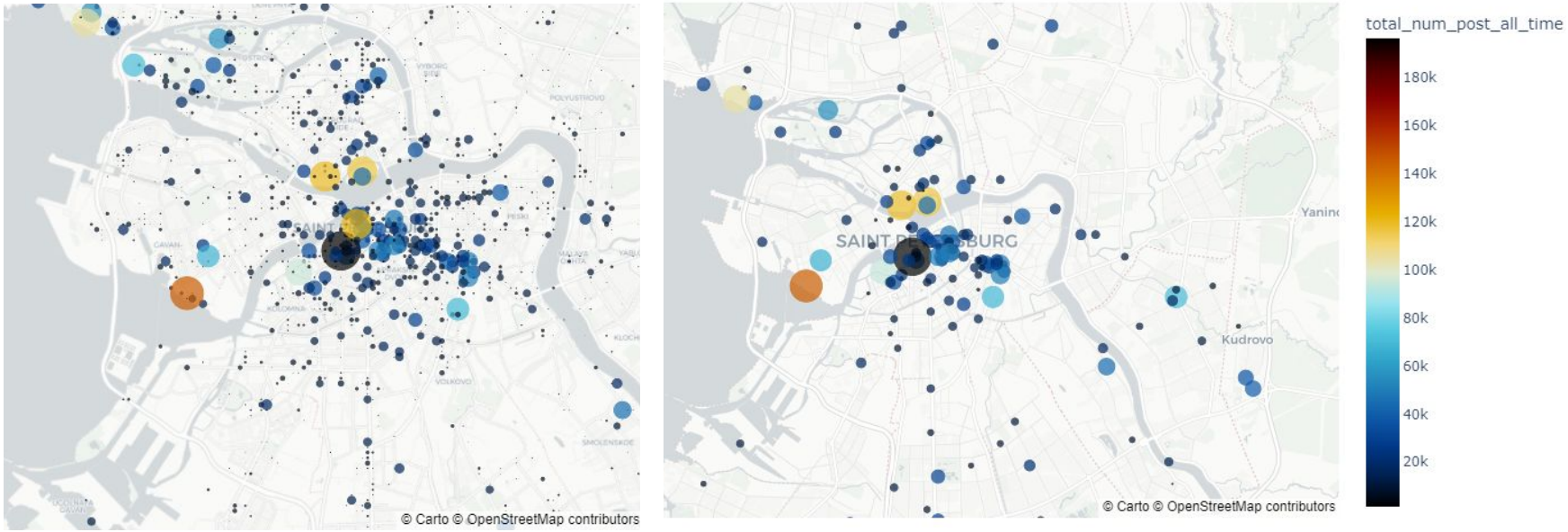


Figure 3 - Difference between train and validation datasets

Final Train Dataset

Data preprocessing steps:

- Filtering by latitude and longitude
- Converting timestamps into regular date form
- Grouping by point - hour groups and calculating total number of publications for all groups
- Filling out passes in point-hour groups
- Then since further we need to estimate quality of model for validation data, we found intersection between points from train and validation datasets, and dropped points that were not in valid data.

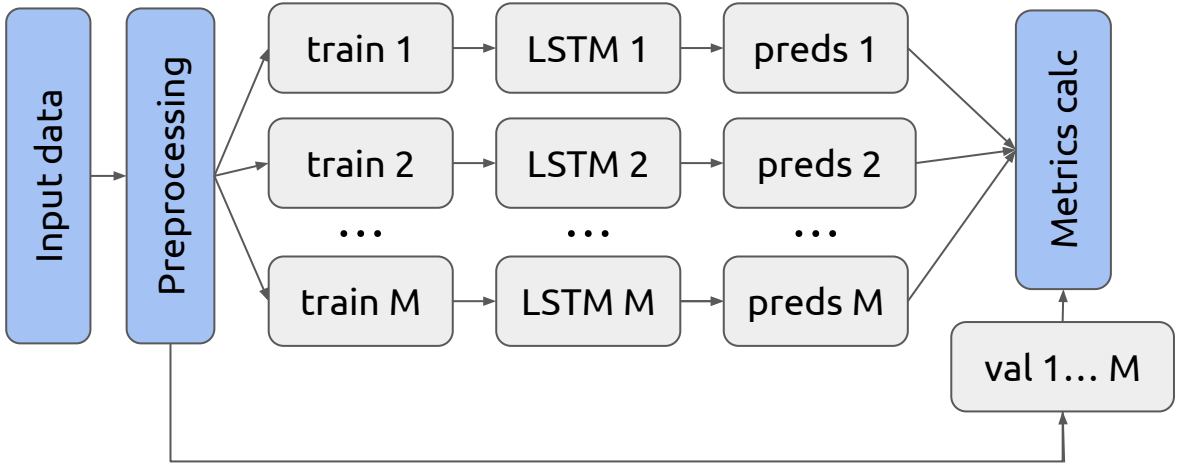
After all preprocessing steps we had following structure of training and validation dataframes:

	date	n_posts
2019-01-01 00:00:00		2.0
2019-01-01 01:00:00		0.0
2019-01-01 02:00:00		0.0
2019-01-01 03:00:00		0.0
2019-01-01 04:00:00		0.0
...		...
2020-01-31 19:00:00		3.0
2020-01-31 20:00:00		7.0
2020-01-31 21:00:00		5.0
2020-01-31 22:00:00		3.0
2020-01-31 23:00:00		3.0

Figure 4 - Train and Val dataframes structure

Model Choice and Description

For current task we have decided to use LSTM model for time series data processing.



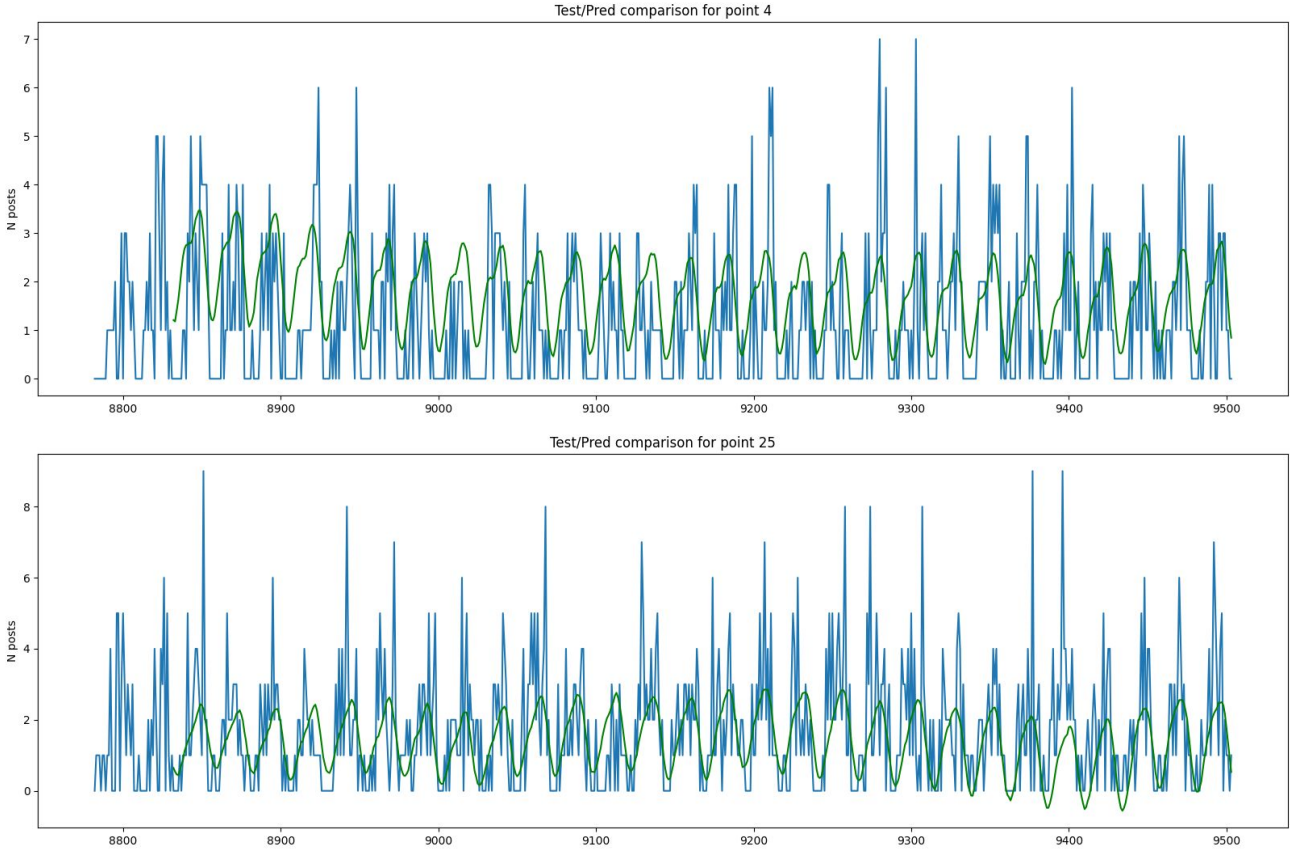

```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape
=================================================================
 lstm_2 (LSTM)               (1, 12)

 dense_2 (Dense)             (1, 672)

=================================================================
Total params: 9936 (38.81 KB)
Trainable params: 9936 (38.81 KB)
Non-trainable params: 0 (0.00 Byte)
```

Figure 5 - Model architecture and Layers description

Model Params:

n_lag = 12 | n_seq = 24 * 28 | n_test = 2 | n_epochs = 2 | n_batch = 1 | n_neurons = 12

Prediction Example



Blue line - True values
Green line - predictions

Figure 6 - Model prediction examples for points number 4 and 25

Errors Distribution

Histogram and stats for errors

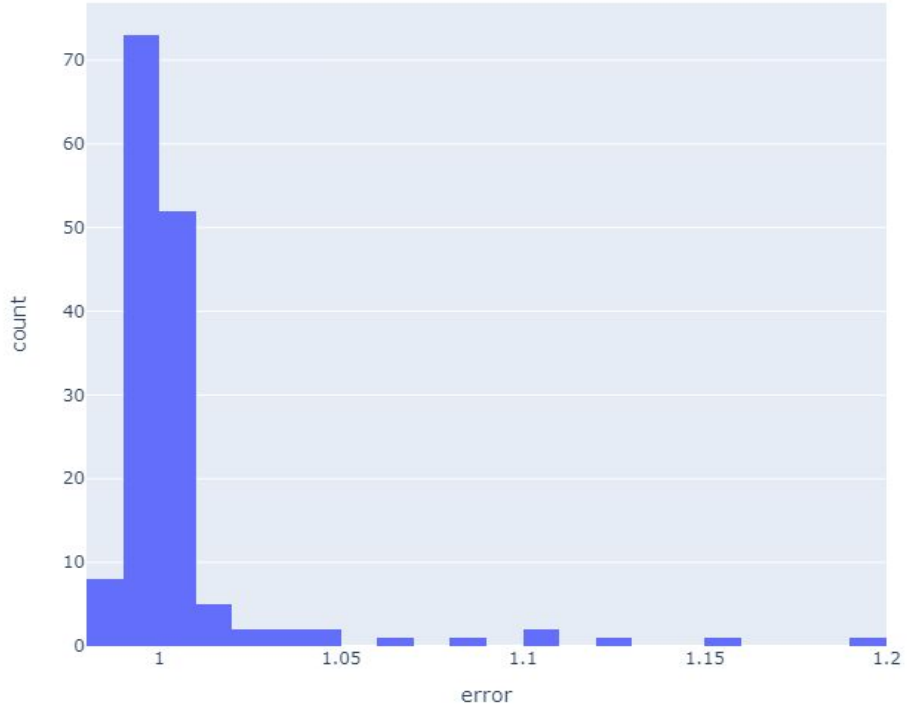


Figure 7 - Errors distribution over the validation dataset

```
count    151.000000
mean       1.006504
std        0.027963
min        0.985524
25%        0.997268
50%        0.999497
75%        1.003614
max        1.195492
dtype: float64
```

Figure 8 - Stats for errors

Thank you for attention!

Google Colab notebook:
https://colab.research.google.com/drive/1HPzacbL9Dt52KKqz-epo6QiRgTt-bPuj?usp=sharing#scrollTo=yUBAa3cygRSi&uniqifier=3