



Cold Start Energy Forecasting

First prize solution

guillermobarbadillo@gmail.com

1. Challenge description
2. Data exploration
3. First steps
4. Tailor made NN
5. Seq2Seq
6. Team

1. Challenge description

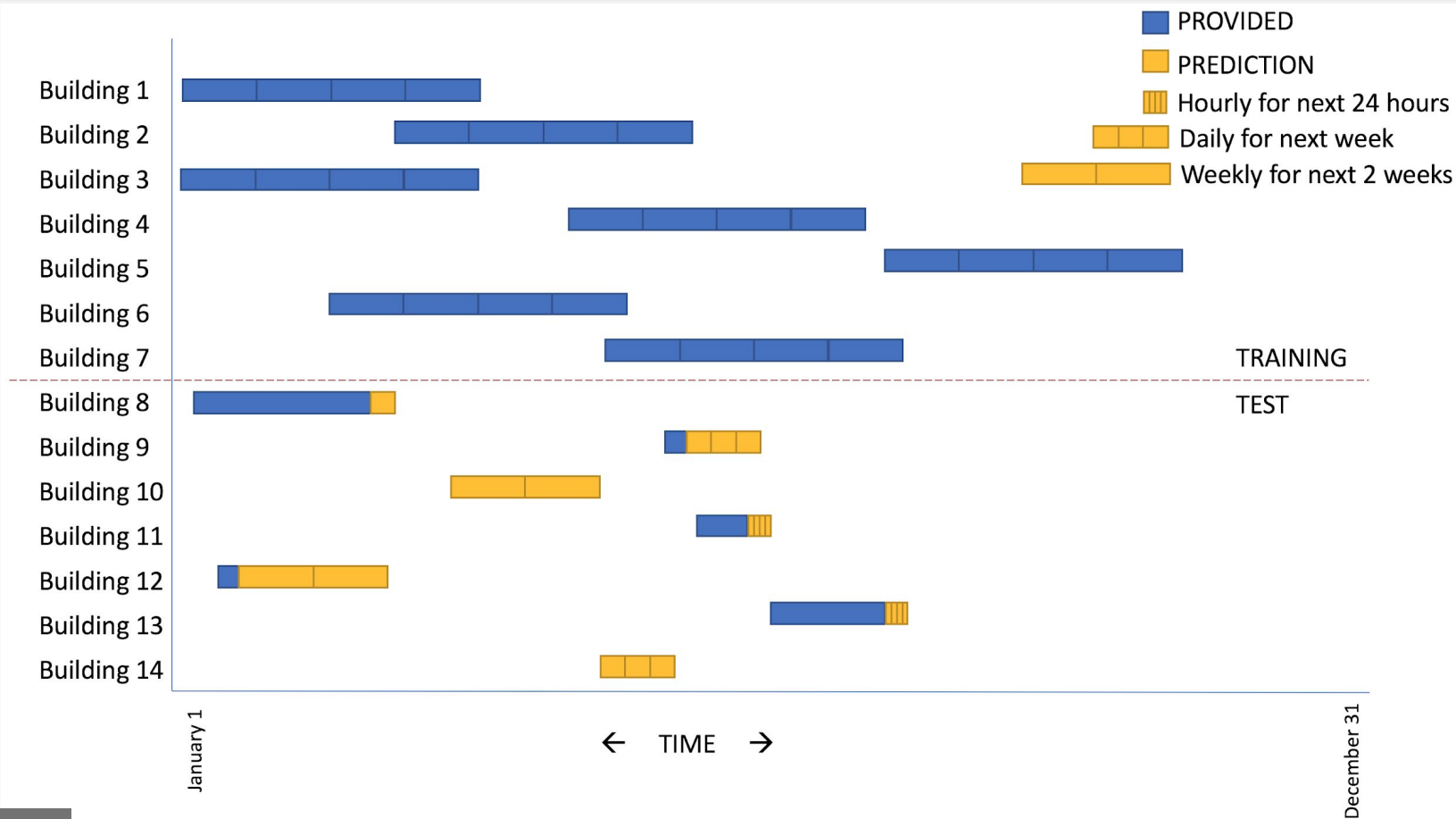
Challenge description

The objective of this competition is to **forecast energy consumption** from varying amounts of "cold start" data, and little other building information. That means that for each building in the test set you are given a small amount of data and then asked to predict into the future.

The "cold" refers to having small amount of data. In the worst case we only have one day of input and we have to make a prediction for 2 weeks.



Challenge description



Challenge description

Three time horizons for predictions are distinguished. The goal is either:

- To forecast the consumption for each hour for a day (24 predictions).
- To forecast the consumption for each day for a week (7 predictions).
- To forecast the consumption for each week for two weeks (2 predictions).

And we can have from 1 day to 14 days of initial data.

Evaluation

The performance metric is a normalized version of mean absolute error.

$$NMAE = \frac{1}{N} \sum_{i=1}^N |\hat{y}_i - y_i| c_i$$

- N - The total number of consumption predictions submitted, including all hourly, daily, and weekly predictions
- \hat{y}_i - The predicted consumption value
- y_i - The actual consumption value
- c_i - The normalization coefficient that weights and scales each prediction to have the same impact on the metric

The normalization coefficient c_i for the i^{th} prediction is composed of a ratio of two numbers,

$$c_i = \frac{w_i}{m_i}$$

- w_i is a weight that makes weekly (24 / 2), daily (24 / 7), and hourly (24 / 24) predictions equally important. This means that weekly predictions are 12 more important than hourly.
- m_i is the true mean consumption over the prediction window under consideration (this mean is unknown to competitors). As far I understand for hourly prediction the mean of the 24 hours is used, for daily prediction the mean of the 7 days and for weekly prediction the mean of the 2 weeks.

Multiplying predictions by this coefficient makes each prediction equally important and puts hourly, daily, and weekly predictions on the same scale.

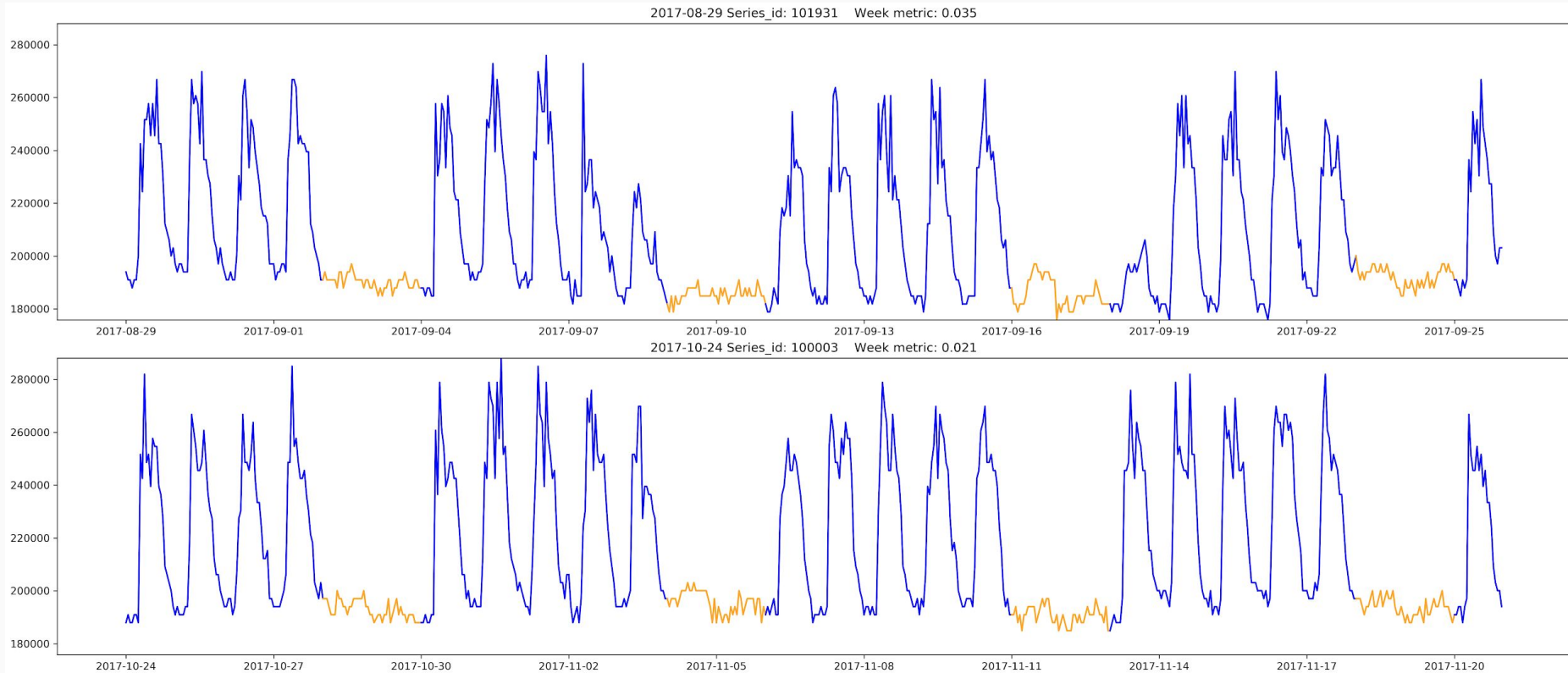
Reference numbers for evaluation

- Hourly: 0.17-0.23
- Daily: 0.13-0.16
- Weekly: 0.11-0.15

When combining those 3 windows we have a score of around 0.30

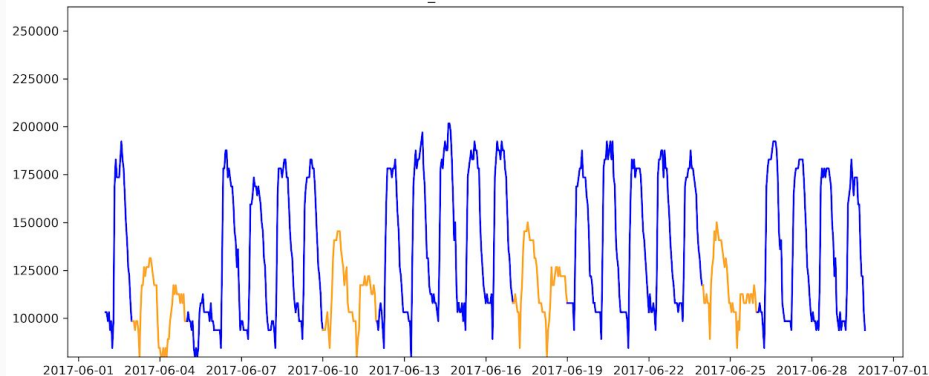
2. Data Exploration

Some buildings are very repetitive

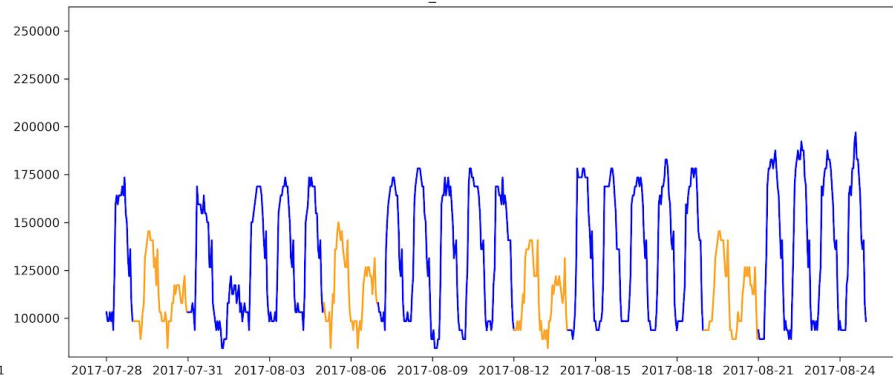


Some buildings are very repetitive

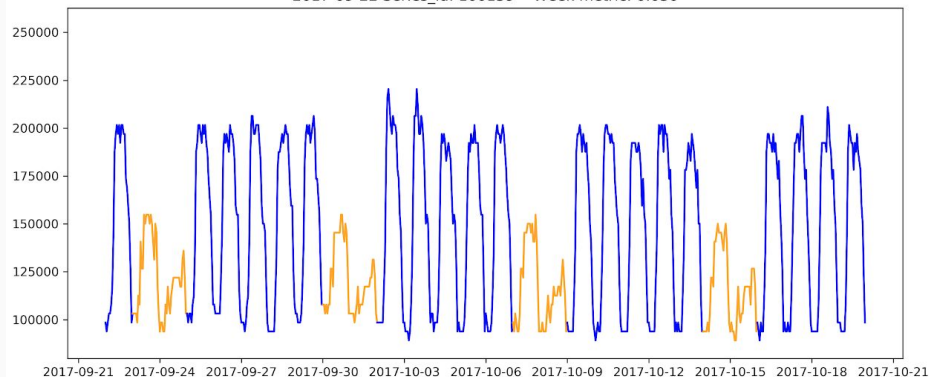
2017-06-02 Series_id: 102521 Week metric: 0.062



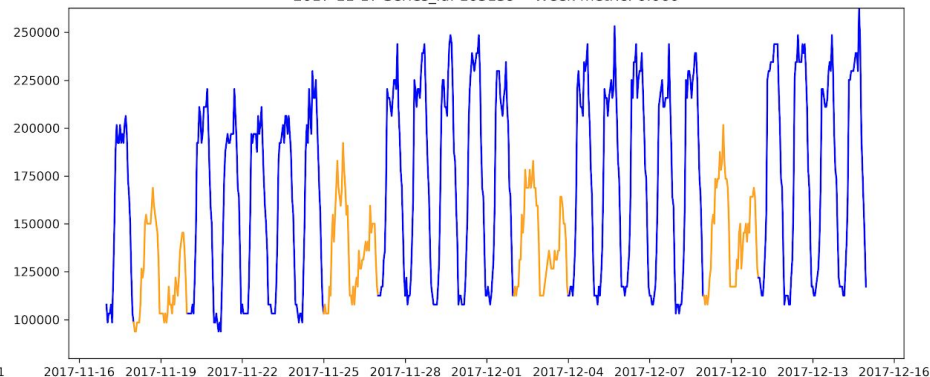
2017-07-28 Series_id: 102709 Week metric: 0.051



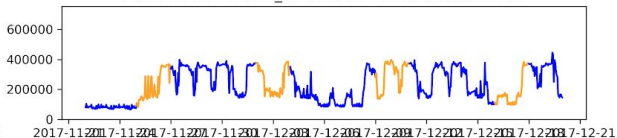
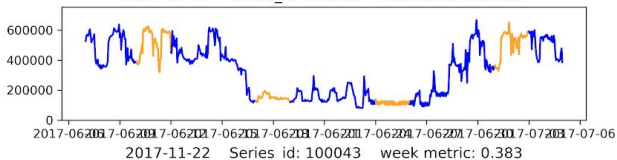
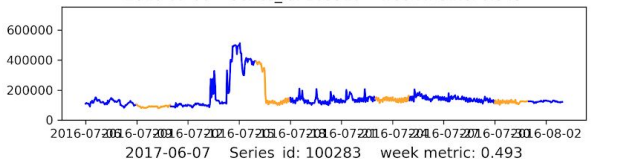
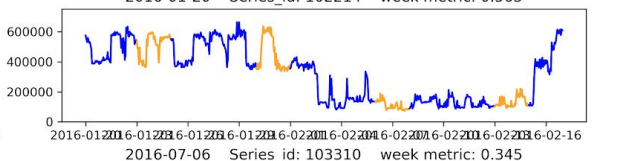
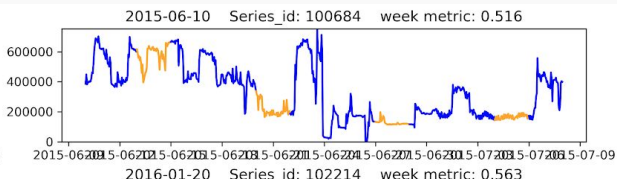
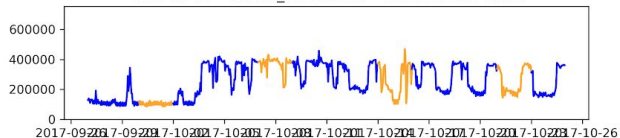
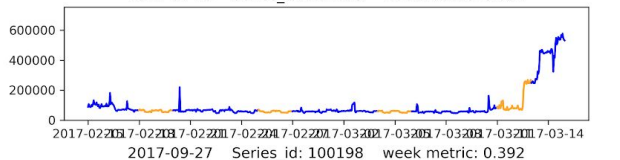
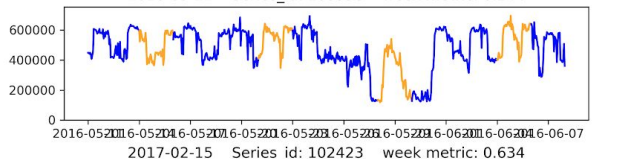
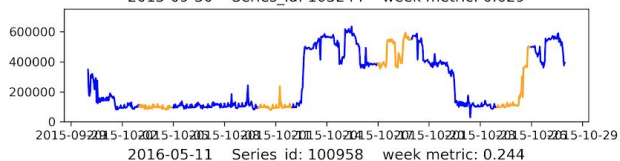
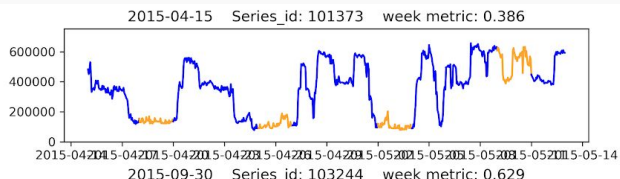
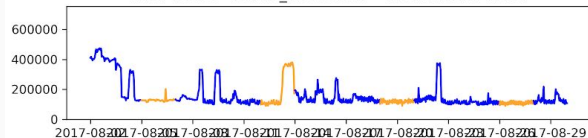
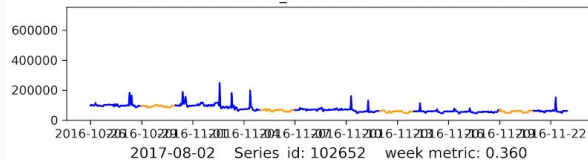
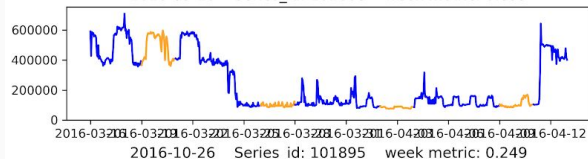
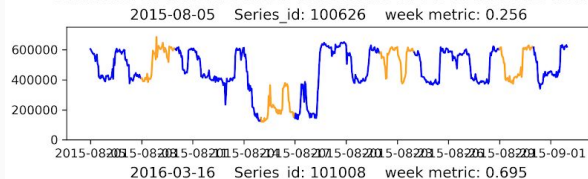
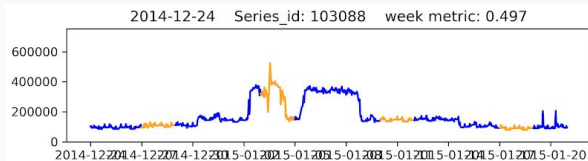
2017-09-22 Series_id: 100139 Week metric: 0.030



2017-11-17 Series_id: 103139 Week metric: 0.060



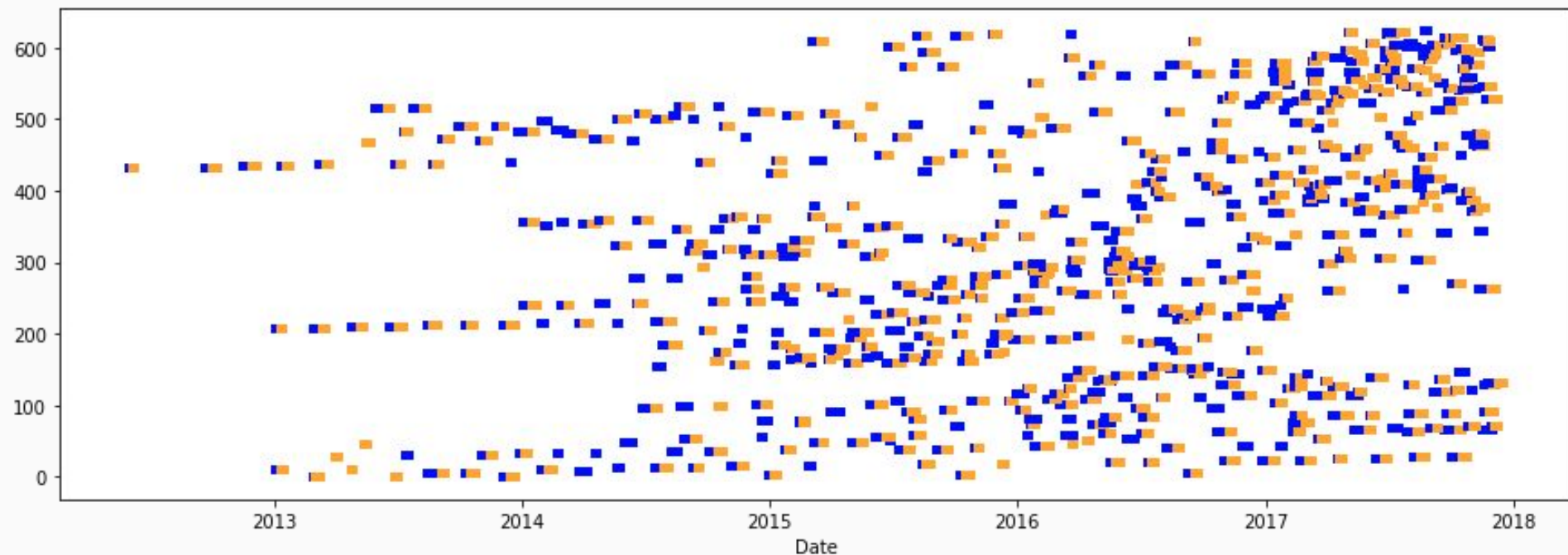
But others are very random...



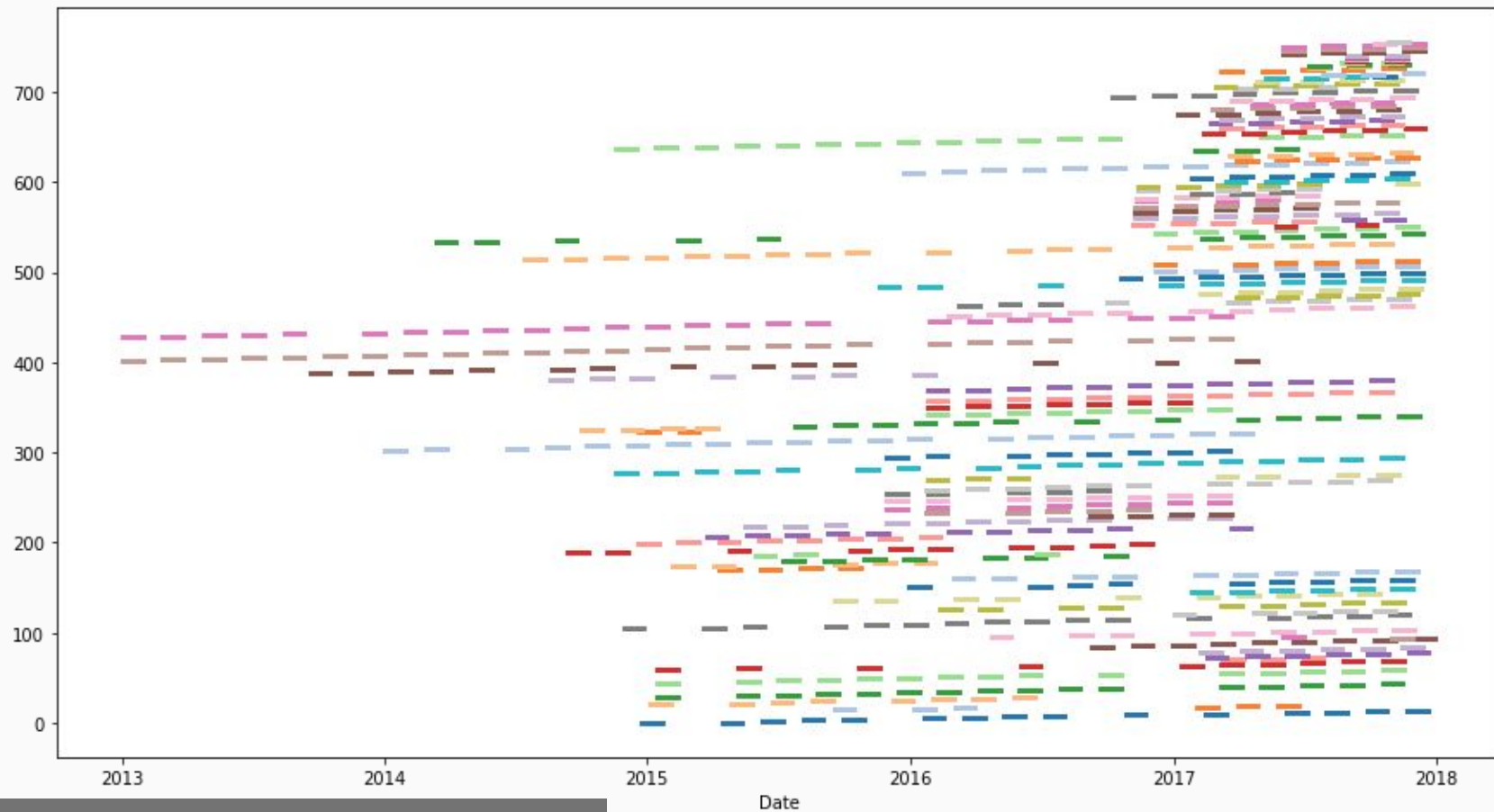
The data

- 758 buildings with 28 days on train set
- 625 buildings on test dataset with different number of days (1-14)
- For each building we have consumption, date and temperature. However 40% of temperature are missing so they were not used on the challenge
- We also have metadata of the buildings: which days are off, size of the building and base temperature

Can you see something strange in this plot?



It should be clearer in this other one



Clusters of buildings

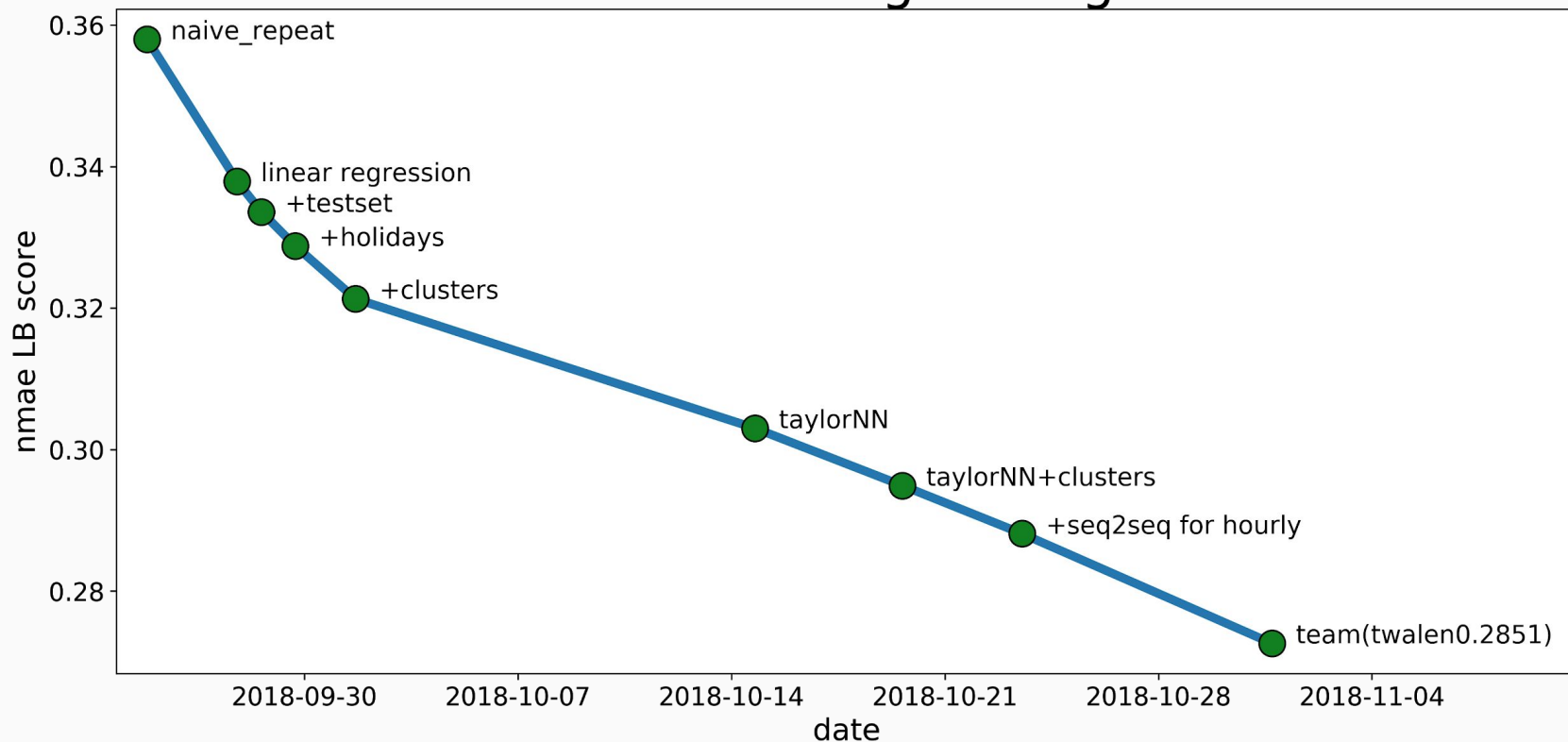
The exploration of the data has revealed that there are less buildings than we thought.

This will be probably be relevant in the challenge because if we only have one day of input but we now that it belongs to a cluster we can use more data for prediction.

3. First steps

The challenge at a glance

Coldstart challenge at a glance



Naive repetition

I wanted to create a baseline that just repeated past data for making predictions.

It achieved much more better scores than the challenge baseline that used LSTM (0.52 vs 0.35)

This simple approach made me realize that there was a very clear difference between working days and days off. I also saw that the further we went in time the more difference between days. It's better to use yesterday than 4 days ago for predicting today's consumption.

Linear regression

After seeing the complex plots of the consumption it seemed to me that predicting them was very difficult. So instead of predicting the consumption from scratch I decided that I will try to weight the past data to create the future one.

For example if I have two days of input I can predict the future simply by averaging them. Or I can give 0.6 weight to yesterday and 0.4 to the day before. Those weights are optimized to reduce the error.

Linear regression

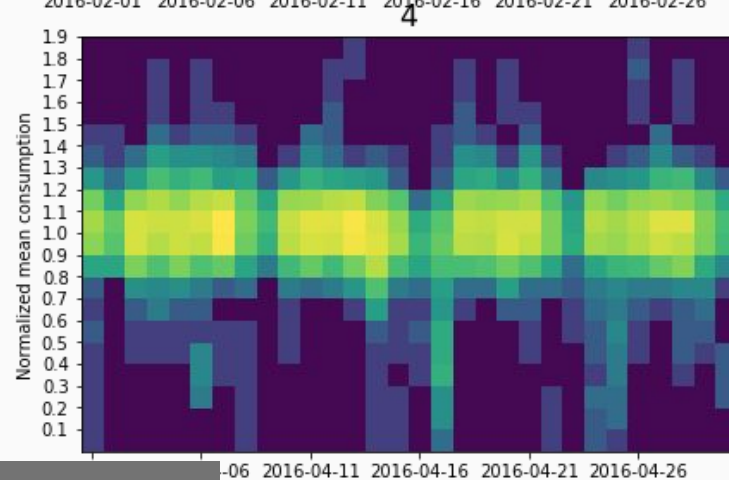
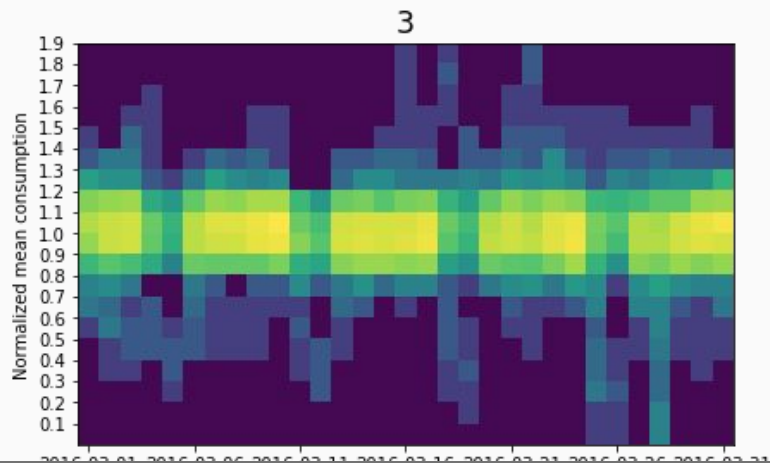
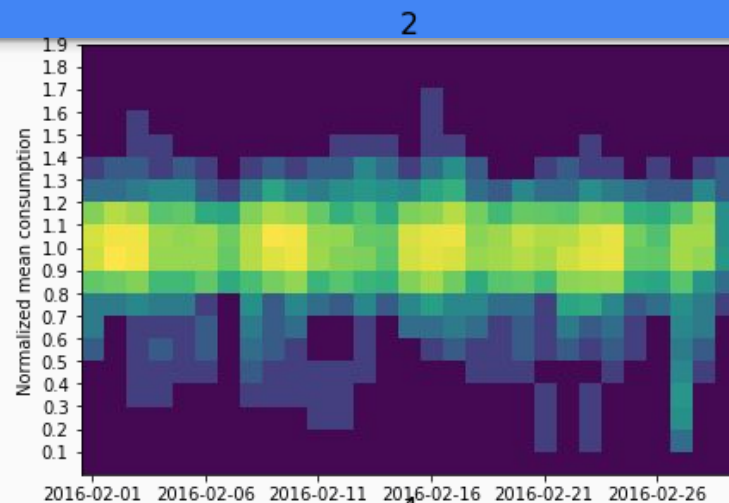
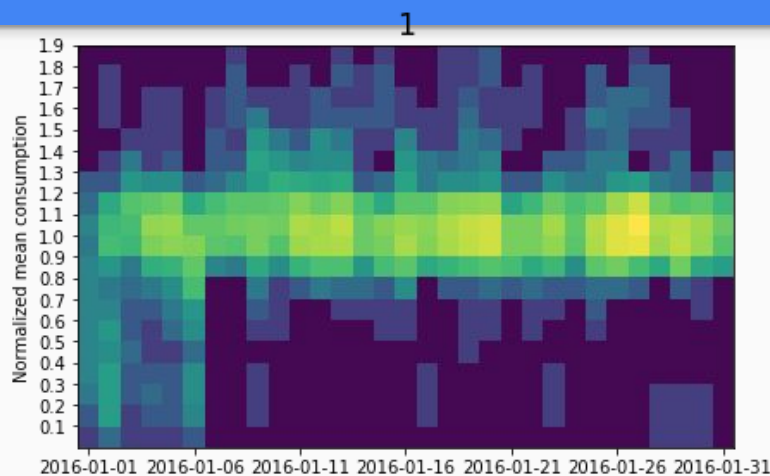
I divided the problem into subproblems using the `is_day_off` vector. For example having two days off as input and having to predict a working day is encoded as 110. I took all cases on train set with that encoding and found the weights that minimize the error of prediction.

Learnings from linear regression

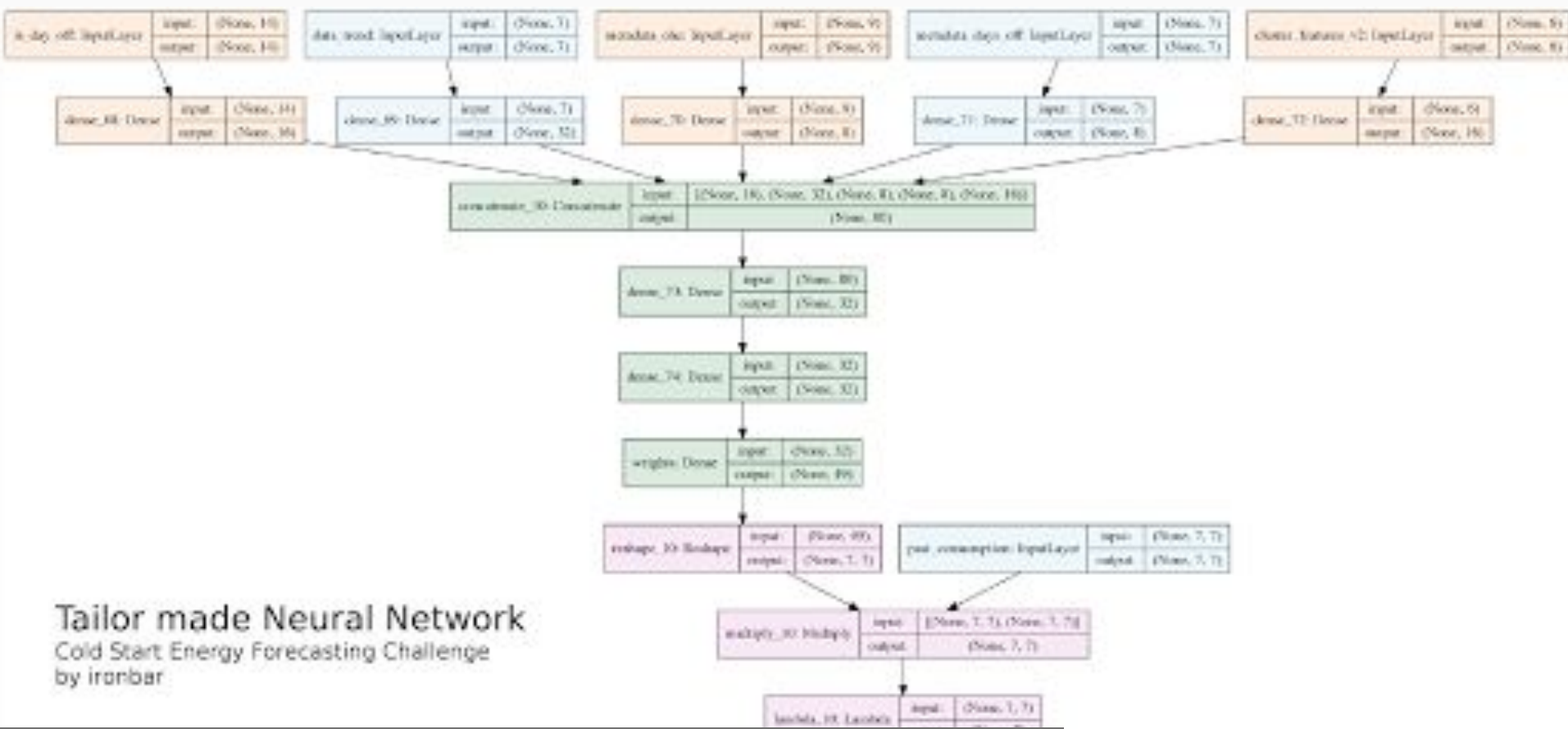
- Using the test data for training improves LB score
- Using an external set of holidays improves LB score
- Using cluster information improves LB score

However this simple model have drawbacks. The only input it was receiving was days-off information. So we needed a more complex model to improve the scores.

Finding holidays on the dataset



4. Tailor made NN



Tailor made Neural Network
Cold Start Energy Forecasting Challenge
by ironbar

Tailor made NN

This model creates an encoding for each of the inputs. After that concatenates all the encodings and predicts the weights for combining the past consumption to create the future consumption.

Having a separated encoding for each input allows to control the importance of each feature and to avoid overfitting.

One great advantage of this model is that we do not have to scale the past consumption because it is not used as a feature, it is simply multiplied by the predicted weights.

Convergence when training this model was not easy and we had to use small batch sizes (8) because using bigger batches lead to bad results. Also gradient clipping was necessary.

Features

- **days off** There is a clear separation between working days and days off. At the start of the challenge this was my only feature and I was able to reach 3rd position using this feature and linear regression only. Using holidays was beneficial.
- **consumption trend** Sometimes there is a clear decrease or increase tendency in the consumption, so giving the trend as input helps to improve.
- **metadata** Encoding the size of the building and base temperature with one hot encoding also helps to make better predictions.
- **cluster_features** I have found that there are clusters of buildings on the data. It greatly improves the scores of predictions

Number of models

A minimum number of 21 models were trained. Different models were trained for each time window and for the number of input days. I realized that using more than 7 days of input did not improve the scores. So $3 \times 7 = 21$

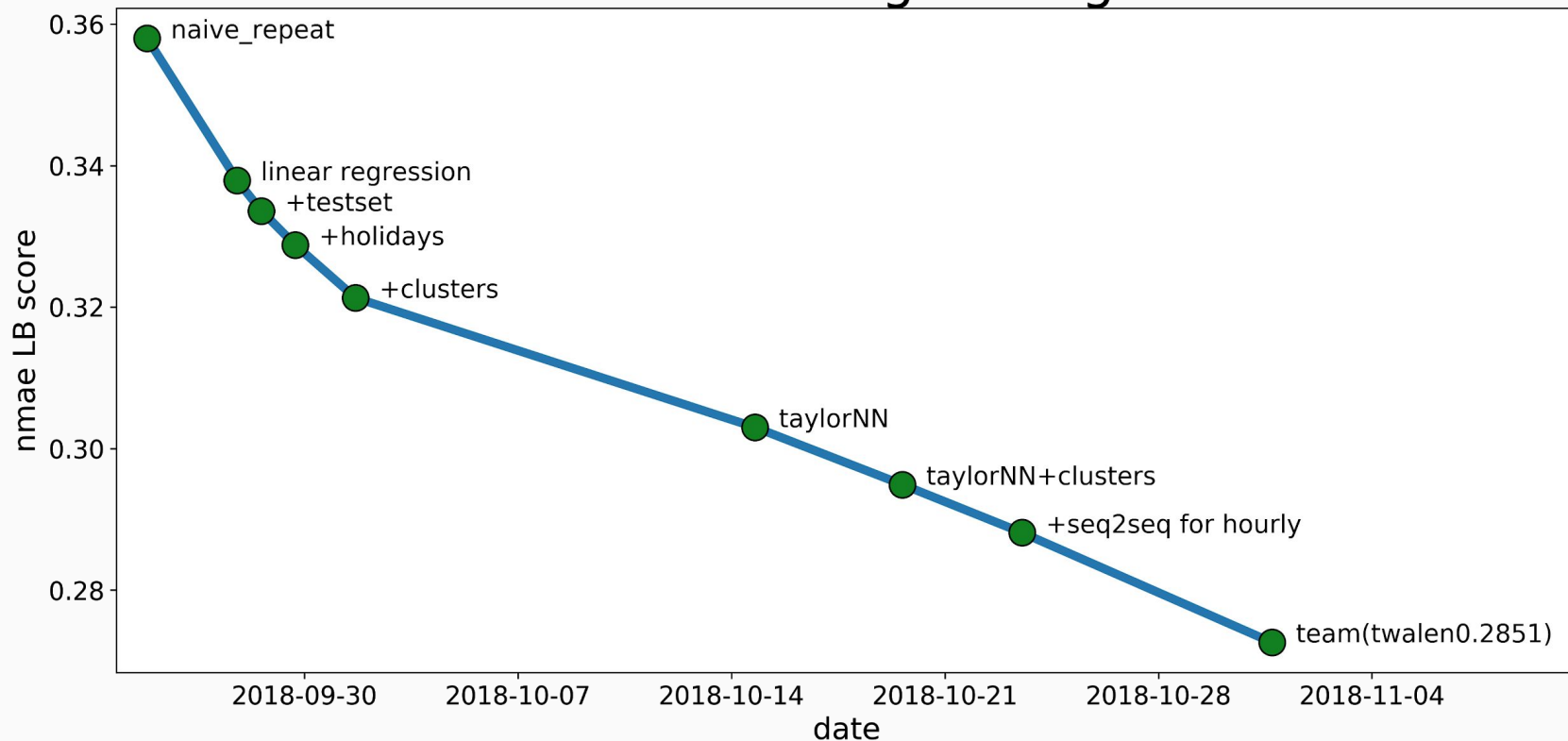
In the final solution I used 5 sets of 21 models (105 models) and I tried an even bigger ensemble with more than 30 sets of models but did not improve the score.

How to use cluster information?

- One hot encoding. Worked well for train set but not for test because there is less data
- Computing cluster features. I computed metrics for each cluster that measured similarity between working days, similarity between days off, similarity between same day on different weeks... This metrics were also a description of the cluster and generalized better

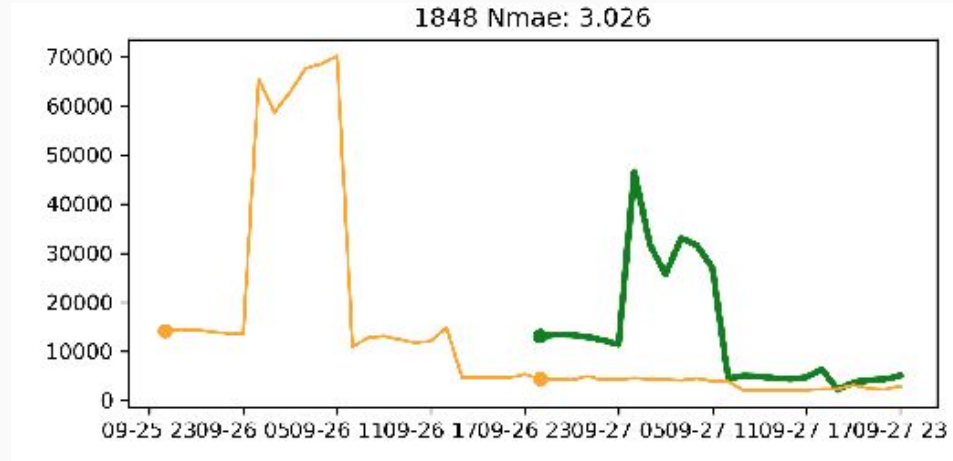
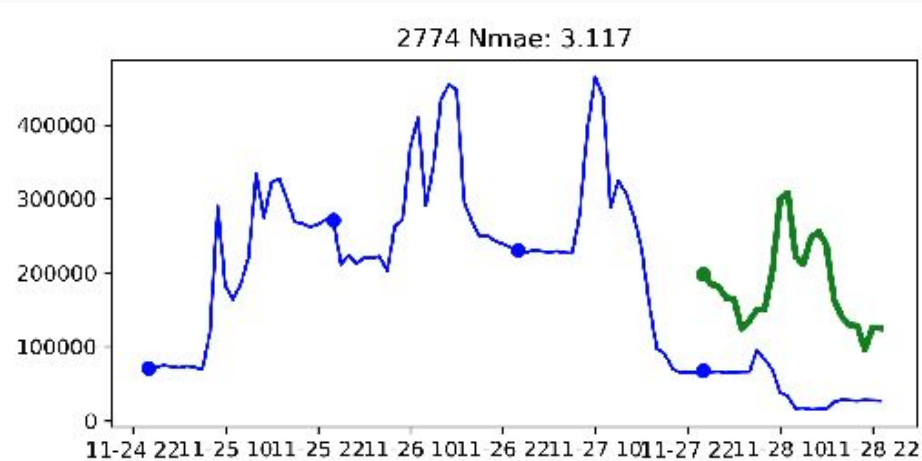
The challenge at a glance

Coldstart challenge at a glance



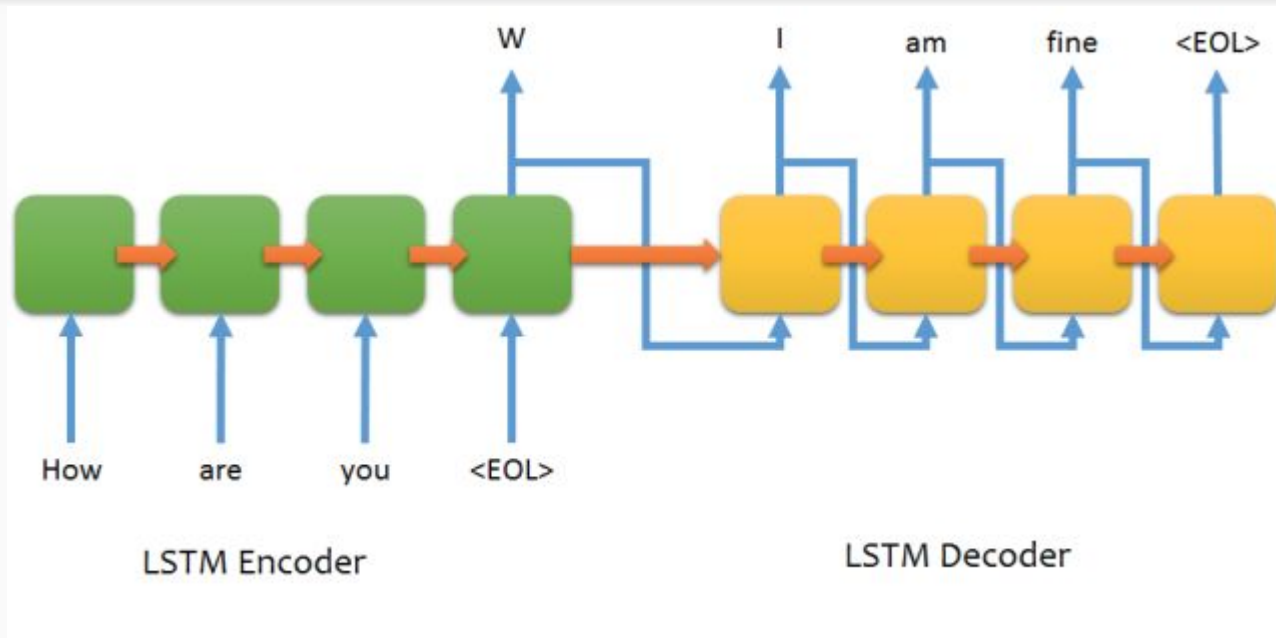
Drawback of Tailor made NN

It did not learn to connect the end of the last day to the start of the new day.



5. Seq2Seq

Seq2Seq



This models are very interesting but have one drawback. We have to give as input all the context needed for making the prediction.

This is not true for our problem because when making the predictions we do not rely only on past data. We also need to know if the next day is a working day or is a day off.

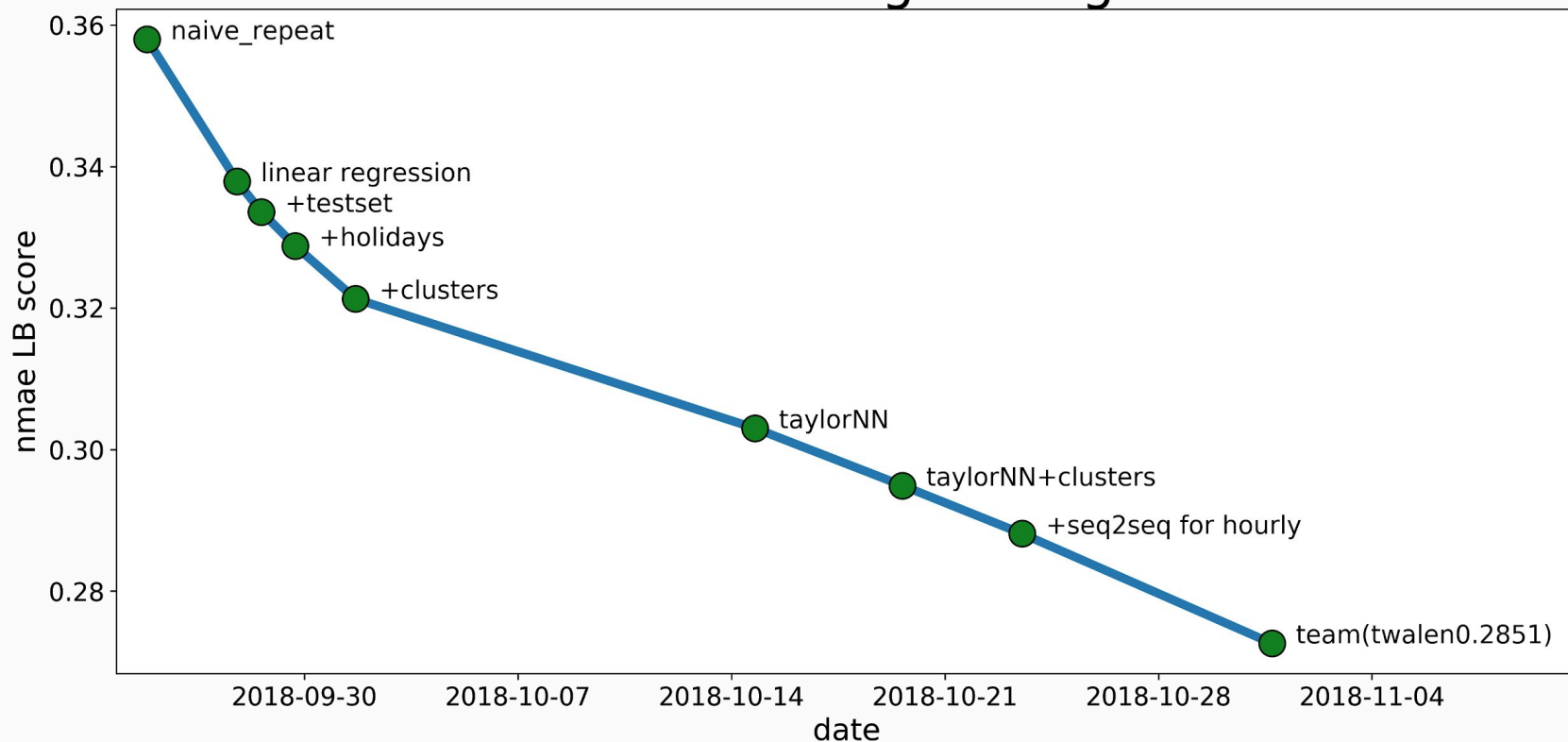
This caused that I could only used this model for predicting hourly. That way we are only predicting for one day. What I did is to train one model for days off and another model for working days. That solved the context problem.

Frankenstein architecture

I tried to create a frankenstein model that combined goodness of seq2seq and used more info for prediction but it did not worked better on LB (although it worked well on validation). This should have worked better because some features are better processed by LSTM and others by vanilla networks (time series and cluster features).

The challenge at a glance



Coldstart challenge at a glance























6. Team

Team with Tomasz

When we team up we were 3 and 4. We had scores of 0.2851 and 0.2881. Combining them we were able to improve to 0.2726. That is the power of ensembles.

User or team		Best public score ⓘ	Timestamp ⓘ	Trend (last 10)	# Entries	
	DenisVorotyntsev	1	0.2775	2018-10-27 06:32:39		66
	valilenk	2	0.2806	2018-10-28 12:59:23		86
	twalen	3	0.2851	2018-10-24 11:56:07		75
	ironbar	4	0.2881	2018-10-27 07:34:56		47

Final LB

User or team		Best private score ⓘ	Timestamp ⓘ	Trend (last 10)	# Entries
 last_minute_team	1	0.2578	2018-10-31 17:18:01		133
 valilenk	2	0.2597	2018-10-27 11:34:22		92
 LastRocky	3	0.2615	2018-10-24 15:14:09		44
 DenisVorotyntsev	4	0.2641	2018-10-25 12:07:46		72
 Li-Der	5	0.2733	2018-10-09 09:05:04		75
 Oneday	6	0.2758	2018-10-31 07:01:48		85
 tanxiao	7	0.2799	2018-10-31 01:59:17		7
 Holberg AS	8	0.2829	2018-10-22 20:40:44		101
 davebel	9	0.2862	2018-10-31 09:26:56		23
 linglu	10	0.2864	2018-10-30 08:04:31		4

Usually with buildings, bigger the historic datasets yield more accurate consumption forecasts. The goal of this challenge is to provide an accurate forecast from the very beginning of the building instrumentation life, without much consumption history.

Quick Facts

PARTICIPANTS 1,291

NO. OF ENTRIES 3,141

PRIZE €23,000

WINNER



last_minute_te...
1ST PLACE TEAM

LEADERBOARD RESULTS

Thank you!



Links

<https://www.drivendata.org/competitions/55/schneider-cold-start/>

<https://github.com/farizrahman4u/seq2seq>