Final Report: Weather Forecast

Advanced Data Mining (CS573)

Students: Mohamad Aljazaery, Ahmed Cheikh Sidiya, Mohammad Reza Khalghani

Spring 2016

Outline:

1) Introduction

   -Problem definition

   - The proposed methodology

2) Pre-processing the data

   -Handling the missing data

   - Normalization

   -Aggregating the airports (Clustering Method)

3) Experiments for validation

   3.1) Experiments conditions

      Error calculation

   3.2) The results

4) Evaluation

5) Conclusion

6) Future Works

1. Introduction:

Weather forecasting has been one of the most scientifically and technologically challenging problems around the world in the last century. This is due mainly to two factors: first, it's used for many human activities and secondly, due to the opportunism created by the various technological advances that are directly related to this concrete research field, like the evolution of computation and the improvement in measurement systems [3]. To make an accurate prediction is one of the major challenges facing meteorologist all over the world. Since ancient times, weather prediction has been one of the most interesting and fascinating domain. Scientists have tried to forecast meteorological characteristics using a number of methods, some of these methods being more accurate than others [5].

Weather forecasting entails predicting how the present state of the atmosphere will change. Present weather conditions are obtained by ground observations, observations from ships and aircraft, radiosondes, Doppler radar, and satellites. This information is sent to meteorological centers where the data are collected, analyzed, and made into a variety of charts, maps, and graphs. Modern high-speed computers transfer the many thousands of observations onto surface and upper-air maps. Computers draw the lines on the maps with help from meteorologists, who correct for any errors. A final map is called an analysis. Computers not only draw the maps but predict how the maps will look sometime in the future. The forecasting of weather by computer is known as numerical weather prediction. To predict the weather by numerical means, meteorologists have developed atmospheric models that approximate the atmosphere by using mathematical equations to describe how atmospheric temperature, pressure, and moisture will change over time. The equations are programmed into a computer and data on the present atmospheric conditions are fed into the computer. The computer solves the equations to determine how the different atmospheric variables will change over the next few minutes. The computer repeats this procedure again and again using the output from one cycle as the input for the next cycle.

Data mining, also called Knowledge Discovery in Databases (KDD), is the field of discovering novel and potentially useful information from large amounts of data [10] . In contrast to standard statistical methods, data mining techniques search for interesting information without demanding a priori hypotheses, the kind of patterns that can be discovered depend upon the data mining tasks employed. By and large, there are two types of data mining tasks: descriptive data mining tasks that describe the general properties of the existing data and predictive data mining tasks that attempt to do predictions based on inference on available data. This techniques are often more powerful, flexible, and efficient for exploratory analysis than the statistical techniques [2] . The most commonly used techniques in data mining are: Artificial Neural Networks, Genetic Algorithms, Rule Induction, Nearest Neighbor method, Memory-Based Reasoning, Logistic Regression, Discriminant Analysis and Decision Trees.

1.1. The case study

For this project, eighteen attributes for 749 airports in the USA related to weather forecasting were given. Data mining methodologies must be implemented on these data to forecast the weather. These attributes are demonstrated in Table 1 that is taken the data:

Table 1. The attributes of the data

airport	date	temp_max	temp_min	temp_avg	temp_dep	temp_hdd	temp_cdd	water	Snow	snow_depth
AAF	5/1/2015	85	58	72	2	0	7	0	0	0
AAF	5/2/2015	80	55	68	-3	0	3	0	0	0
AAF	5/3/2015	81	58	70	-1	0	5	0	0	0
AAF	5/4/2015	83	63	73	2	0	8	0	0	0
AAF	5/5/2015	84	65	75	3	0	10	0	0	M

wind_speed_avg	wind_speed_max	wind_dir	sunshine_min	sunshine_percent	sky_cover	weather_type	wind_highest_speed	wind_highest_dir
6.8	17	20	M	M	0	M	22	310
6.3	15	160	M	M	0	M	19	140
6.2	17	150	M	M	0	M	23	150
7.1	16	110	M	M	1	M	20	110
M	M	M	M	M	M	M	M	M

Each of the these attributes are explained in Table 2:

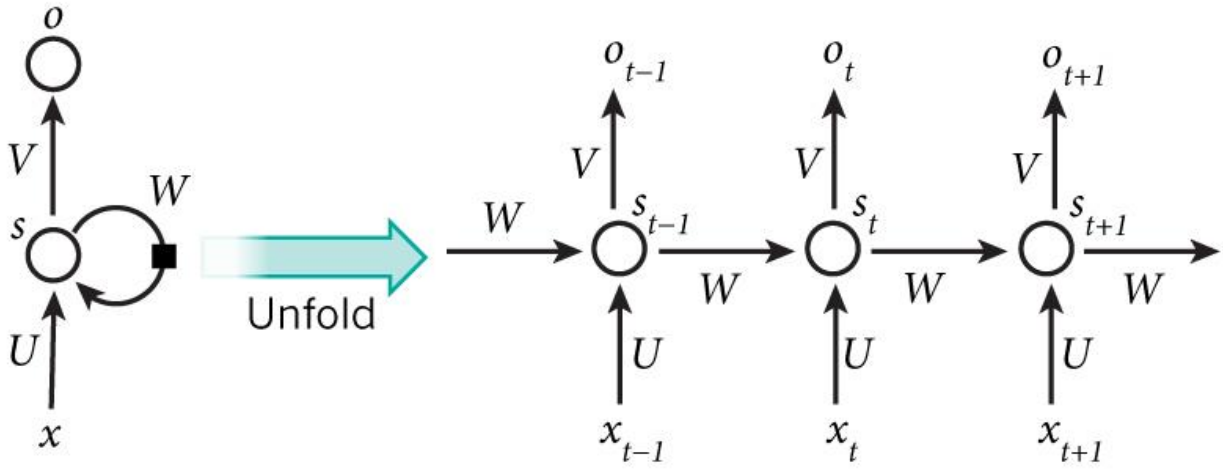
Table 2. Defining all of the attributes [2]

Attributes	Definition
date	The **day** of the month
temp_max	The **highest** temperature for the day in degrees Fahrenheit (F).
temp_min	The **lowest** temperature for the day in degrees Fahrenheit (F).
temp_avg	The **average** temperature for the day, computed by finding the average of the values in columns 2 and 3, then rounding (if necessary). Example; 55.5 rounds up to 56, 55.4 rounds down to 55 degrees.
temp_dep	**Departure** from normal. The difference between column 4 and the 30 year normal temperature for this date. A minus (-) is number of degrees below normal. A zero (0) indicates that the average for that day was the Normal.
temp_hdd	A gauge of the amount of heating or cooling needed for a building using 65 degrees as a baseline. To compute heating/cooling degree-days, take the average temperature for a day and subtract the reference temperature of 65 degrees. If the difference is positive, it is called a "**Cooling Degree Day**".
temp_cdd	If the difference mentioned above is negative, it is called a "**Heating Degree Day**".
water	Total **precipitation** for the day to the nearest hundredth of an inch. This includes all forms of precipitation, both liquid and water equivalent of any snow or ice that occurred

Snow	Total **snowfall** for the day to the nearest tenth of an inch.
snow_depth	Snow **depth** on the ground to the nearest inch at 1200UTC
wind_speed_avg	**Average wind speed** for the day in miles per hour (mph).
wind_speed_max	The **highest wind speed** in mph averaged over a 2 minute period.
wind_dir	The **direction** (in compass degrees divided by 10) from which the wind speed in column 11 came from
sunshine_min	The number of **minutes** of sunshine received at the station. Not reported at all locations.
sunshine_percent	The percentage of **possible** sunshine. Computed by dividing the minutes of sunshine in column 13 by the total possible minutes. Not reported at all locations.
sky_cover	The average sky cover between sunrise and sunset in tenths of sky covered. The minimum of "0" means no clouds observed, "10" means clouds covered the entire sky for that day.
weather_type	A coded number representing certain types of **weather** observed during the day. 1 = Fog 2 = Fog reducing visibility to 1/4 mile or less 3 = Thunder 4 = Ice pellets 5 = Hail 6 = Glaze or rime 7 = Blowing dust or sand: visibility 1/2 mile or less 8 = Smoke or haze 9 = Blowing snow X = Tornado
wind_highest_speed	Peak wind **speed** for the day in mph. The highest wind speed observed at the station.
wind_highest_dir	The compass **direction** from which the peak wind speed came.

1.2. The proposed method

Recurrent Neural Network (RNN):



The idea behind RNNs is to make use of sequential information. In a traditional neural network we assume that all inputs (and all outputs) are independent of each other , so traditional ANN Doesn't learn from the data sequence.

RNNs Also use the backpropagation algorithm, but with a little twist. The gradient at each output depends not only on the calculations of the current time step, but also the previous time steps. For example, in order to calculate the gradient at t=4 we would need to backpropagate 3 steps and sum up the gradients. This is called Backpropagation Through Time (BPTT). Which make it perfect for time series and sequential data problem [4].

In our case, the inputs are the weather features (including mix and max tempreture) for N sequential days. The outputs are the min temperature and max temperature for the same days sequence with a on day shifting to future. In testing we care about the last (min temp,max temp) output which is the N+1 day.

To implement RNN we used Python Theano implementation [5] .

2. Pre-Processing The Data

Data pre-processing is an often neglected but important step in the data mining process, especially in real datasets. Data gathering methods are often loosely controlled, resulting in out-of-range values, impossible data combinations, missing data, etc [3]. If there is much irrelevant and redundant information present or noisy and unreliable data, then knowledge discovery during the training phase is more difficult. Data preparation and filtering steps can take significant amount of processing time. Data preprocessing includes cleaning, normalization, transformation, feature extraction and selection, etc.

Raw data is highly subject to noise, missing values, and inconsistency. The quality of data affects the data mining results. In order to help improve the quality of the data and consequently, of the mining results raw data is pre-processed so as to improve the efficiency and ease of the mining process. This process is one of the most critical steps in data mining procedure that deals with the preparation and transformation of the initial dataset. The process is divided in to data cleaning, data integration, data transformation and data reduction.

In this project, preprocessing the data is so important since the data are extracted from a real dataset that missed many values. These missing data can considerably decrease the prediction performance. In Table 3, the percentages of missing data are mentioned:

Table 3: The percentages of missing data for each attribute

Attributes	temp_max	temp_min	temp_avg	temp_dep	temp_hdd	temp_cdd
Missing Data (%)	0	0	0	8	0	0

Attributes	water	Snow	snow_depth	wind_speed_avg	wind_speed_max	wind_dir

Missing Data (%)	11	24	24	0	0	0

Attributes	sunshine_min	sunshine_percent	sky_cover	weather type	wind highest speed	wind highest dir
Missing Data (%)	99	99	2	51	5	6

In order to handle the missing data, we take an average from the remaining data and then replaced the value with the missing data.

Also, in the dataset, there are different attributes that have various units, including Fahrenheit, inch, miles per hour (mph), mph averaged over a 2 minute period, direction, minute. Therefore, we have to normalize the data for increasing the accuracy. Also, there are three reasons for normalizing the input data for RNN:

1) Increasing the training process

2) Removing the dependence among the measurement units

3) Preventing weights from being overly adjusted due to the possible large magnitudes of measured predictor variable values

Two important normalizing methods were applied for this case, MinMax and Z-score. MinMax has the following formula:

$$v' = \frac{v - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A \quad (1)$$

Where, new_max_A and new_min_A are equal to 1 and -1. We had two reasons for considering -1 as new_min_A. The first one is that inputs that range from +1 to -1 are better for neural network. The second reason is that we had *Temp_Dep* as one of the attributes that has negative value. Furthermore, Z-score normalization method follows the following equation:

$$v' = \frac{v - \bar{A}}{\sigma_A} \tag{2}$$

Aggregation of the data set can improve the outcomes of the data mining process. Therefore, a clustering method based on K-means is applied for dividing all of the data into 20 clusters in terms of geographical position. This method can magnify the data with the other airports close to each other in terms of geographical position, Latitude and Longitude (It is supposed that these airports have similar climate pattern). Hence, Latitude and Longitude are two dimensions for this clustering. The results of this aggregation is shown in Figure 1.

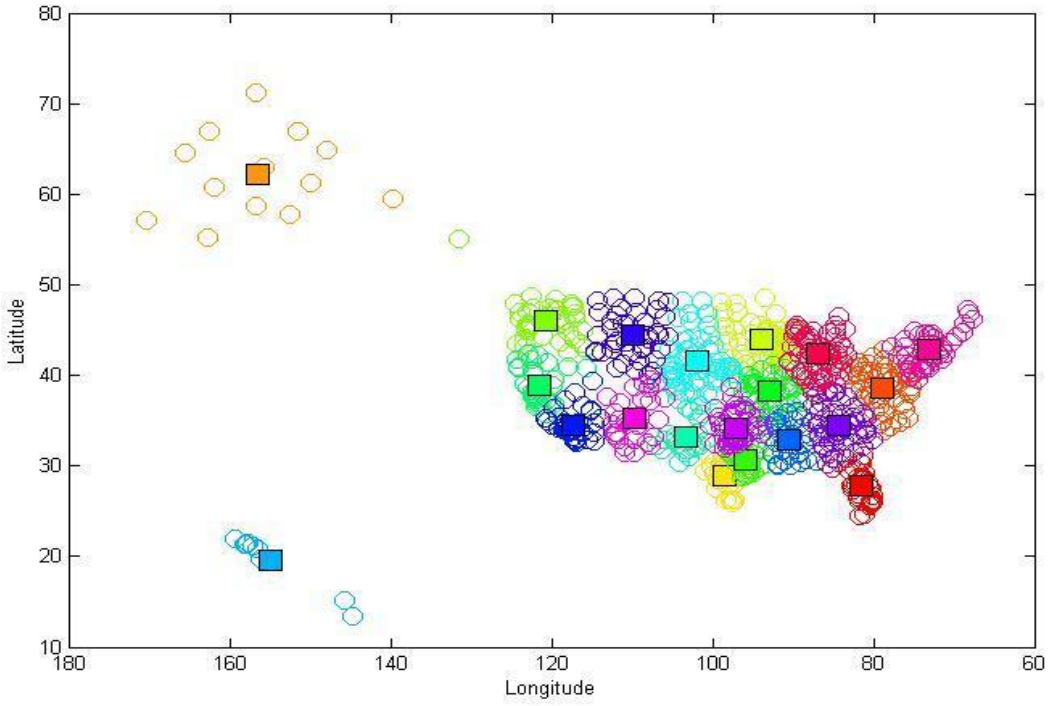


Figure 1. The results of the clustering method (20 clusters of airports based on longitude and latitude)

3. Validation Experiments:

In order to choose the best parameters for our models, multiple validating experiments has been done.

3.1. Experiments condition:

Following parameters are used as a default parameters for All of the experiments:

Fixed parameters:

- Activation function: tanh
- TESTING_PERCENTAGE: 10
- 600 Epoch training
- Hidden Layers : 10
- Normalization: zscore
- Features : All
- Past Days : 7 days
- Data Segmentations: clustering

3.2. Error measurements:

The way we measure the efficiency of our model is by using the mean absolute value. It gives us an error in degree that is easier to interpret.
The formula is the following:

$$Error = \frac{\sum_{i=1}^{N} \left| T_{i,real} - T_{i,predicted} \right|}{N} \quad (3)$$

Experiments number 1: Goal choose the best normalization

Results in the chart below. Z Score is better that Minmax normalization.

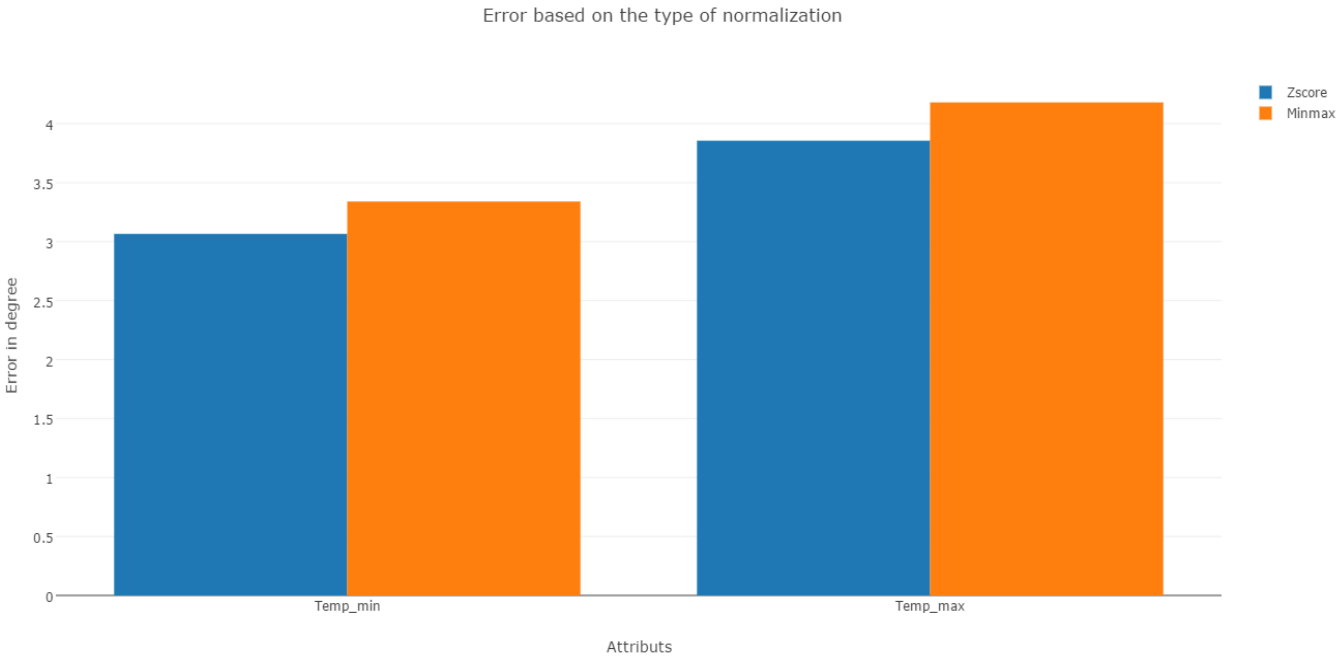


Figure 2. Comparison between two normalization methods

Experiments number 2; Goal finding the best number of hidden layers for RNN.

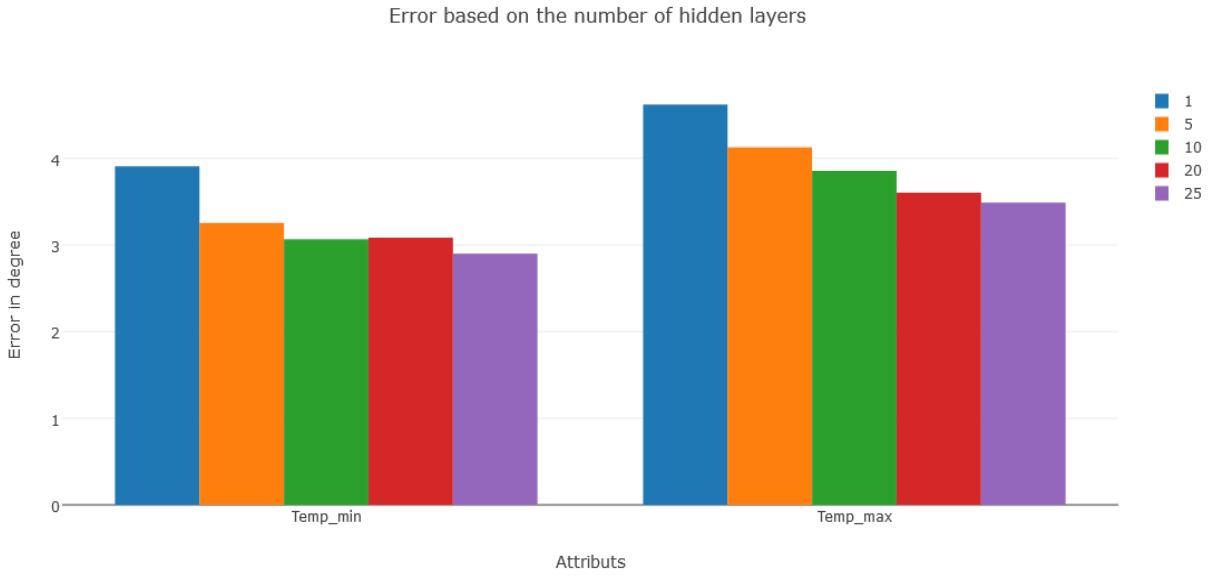


Figure 3. The performance of the prediction based on increasing the number of hidden layers

We can see from the chart below that 25 is the best number of hidden layers

Experiment number 3: Goal is finding the best attributes to select for our model.

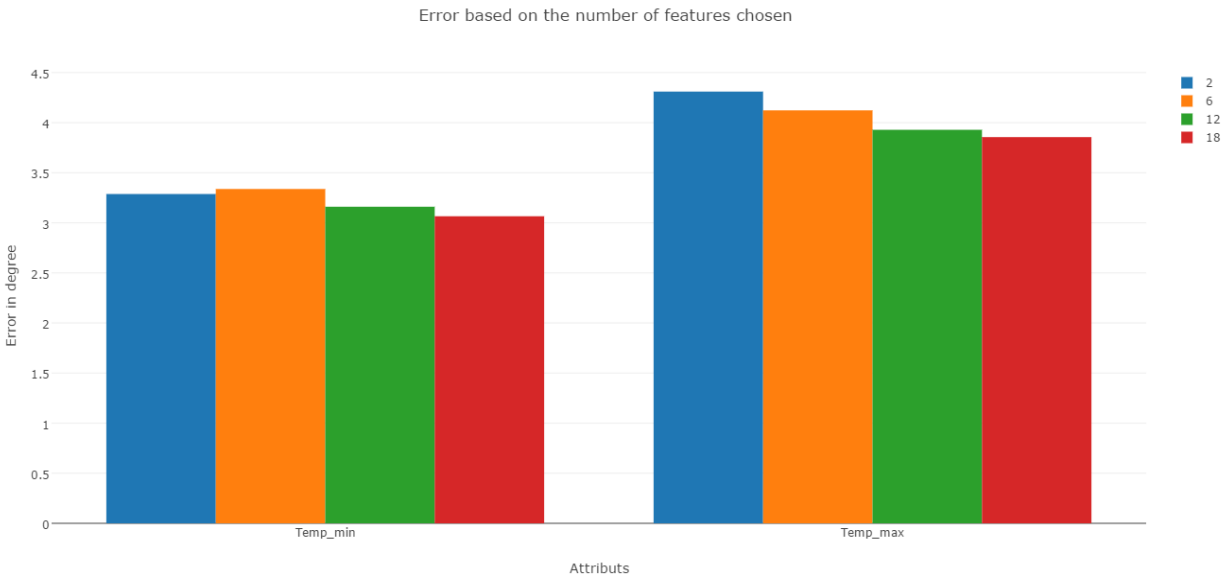


Figure 4. The performance of the prediction based on different number of attributes

We can clearly see in the chart that including all the 18 attributes is the best solution.

Experiment number 4: Goal: Choosing the best number of past days to use for the prediction of the temperature.

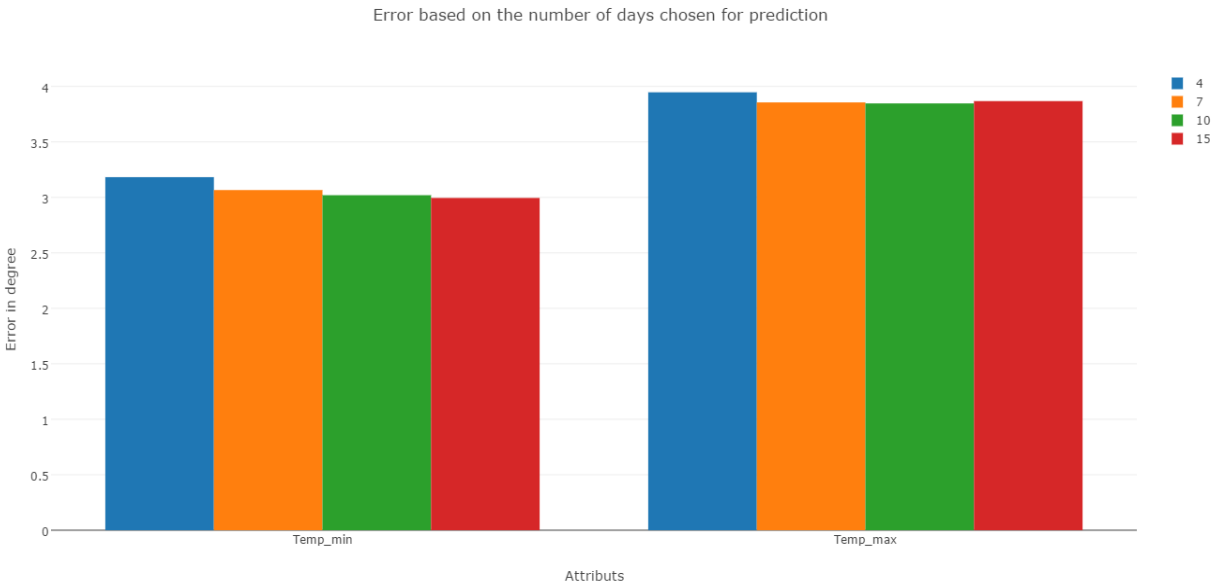


Figure 5. The performance of the prediction based on different number of past days for training

We can see in the chart that the best results are almost between 10, 15 and 7 days. We chose 10 days.

Last Experiment: Goal choosing the best aggregation method for our data.

We can see from that chart below that using independent airport data performs better than clustering or using all the data.

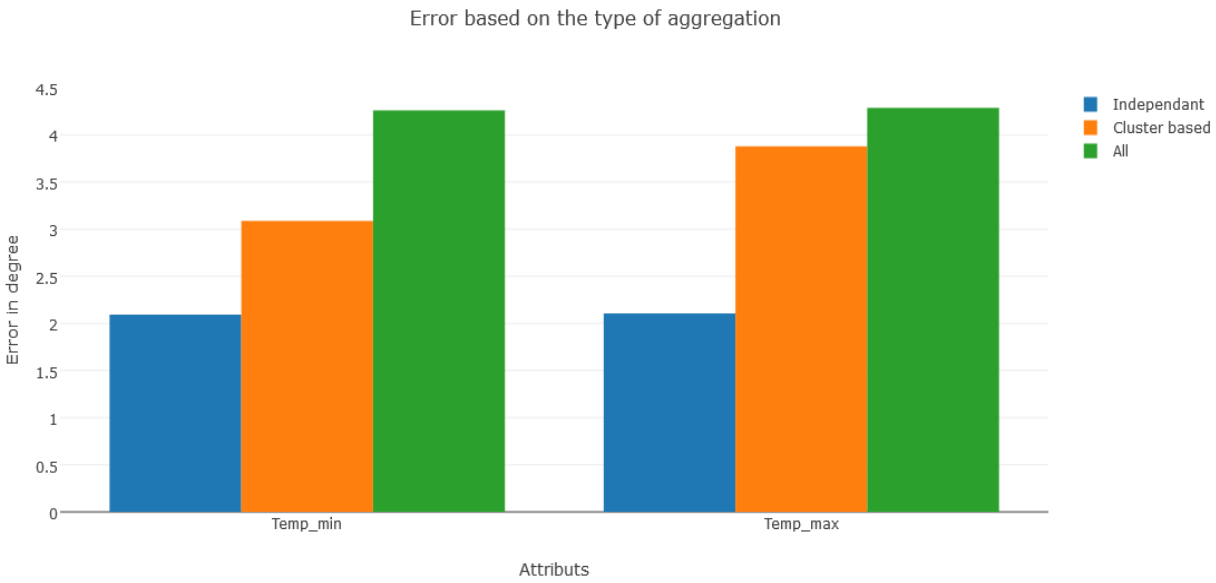


Figure 6. The performance of the prediction based on aggregation conditions

4. Evaluation :

To evaluate our methods, we used linear regression as a baseline. Also, Intellicast forecasting data has been parsed and used in evaluation. For our method, we used the best parameters from the validation experiments and we increased the iterations of the training (Epochs Number) to **600**.

Our Method Attributes:

- ('Hidden Layers ', 25)
- ('PAST DAYS ', 10) : How many days used as a sequence to train the RNN model
- ('TESTING PERCENTAGE ', 10)
- ('Epochs Number ', 600)
- ('Normalization ', 'zscore')
- ALL Data Features: 18 attributes
- Data segmentation: Model for each airport
- Activation function: tanh

Linear Regression Method attributes:

- Input: All features of 10 days as one input vector
- Output: the next day temperature
- Data segmentation: A) Model for each airport B) model using all data

Intellicast:

- Over 100K testing days

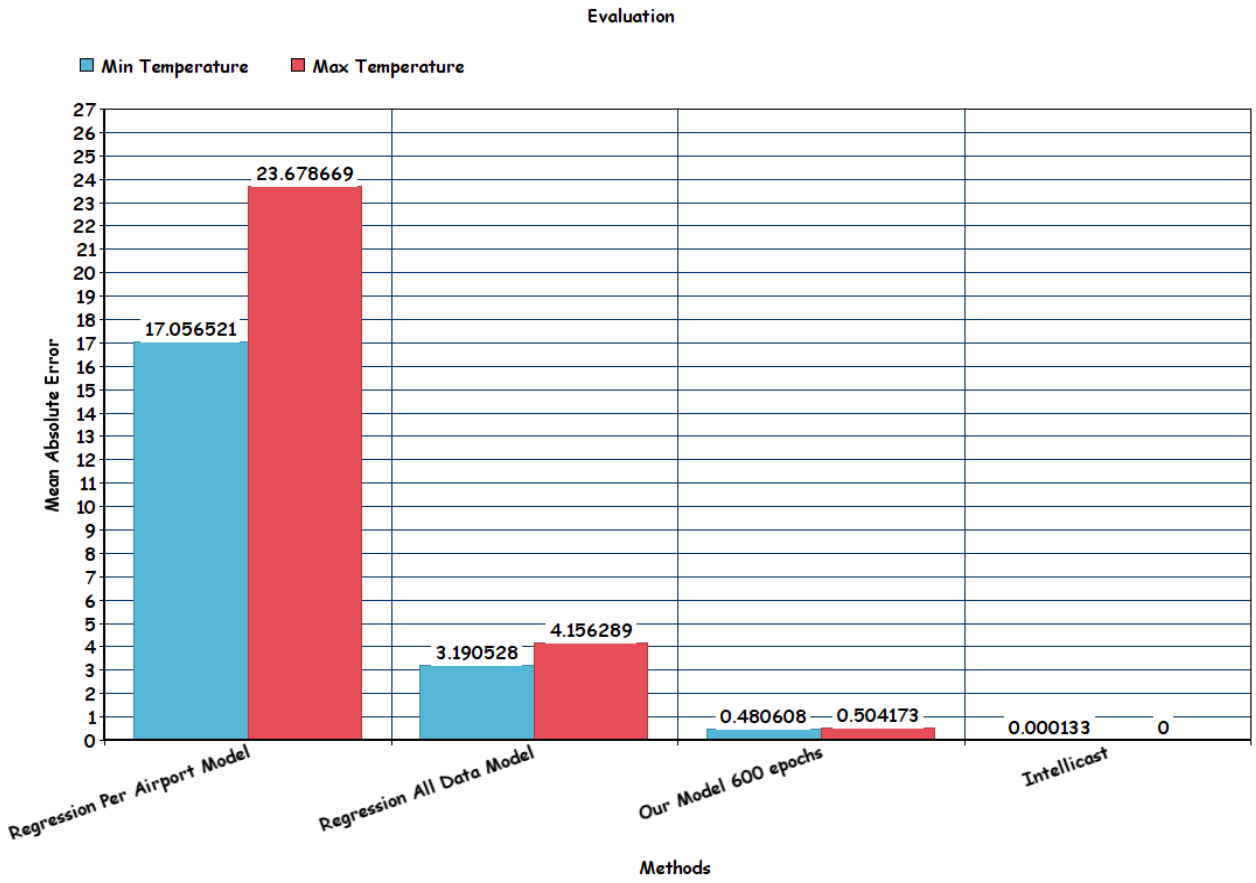


Figure 7. The evaluation of performance among the proposed method and other predictions

As it is examined from the results, RNN has a significantly accurate forecast compared to the base-line method. However, the performance is not as accurate as intellicast prediction.

5. Conclusion:

Our RNN method was much better than linear regression to predict the weather, however the Intellicast forecasting was so professional with almost 0 degree error.

As a summary here is the main points of our conclusion:

- RNN is much much better than Linear regression for time series forecasting

- Zscore normalization is better than minmax for RNN training
- Clustering data into 20 geographical areas based on location only is not helpful. Clustering the airports based on similar climate conditions is better than using location information only.
- Using all the features is helpful, although some features have high missing data percentage.

6. Future Works:

- Use different RNN structures and different activation functions
- Models combination: Try average prediction from two different models (“clustered data” , “independent”)
- Add another dimension to the clustering method (i.e. considering 30 years normal temperature which can be extracted from the data)

References:

[1] Folorunsho Olaiya, "Application of Data Mining Techniques in Weather Prediction and Climate Change Studies" I.J. Information Engineering and Electronic Business, 2012, 1, 51-59.

[2] "National Weather Service Climate." *National Weather Service Climate*. Web. 01 May 2016. [http://w2.weather.gov/climate/f6.php?wfo=gld]

[3] "Data Pre-processing." *Wikipedia*. Wikimedia Foundation. Web. 01 May 2016.

[4] T. Mikolov, M. Kara at, L. Burget, J. Cernocky, and S. Khudanpur. Recurrent neural network based language model.

[5] "Gwtaylor/theano-rnn." *GitHub*. Web. 01 May 2016. [https://github.com/gwtaylor/theano-rnn]

[6] "Recurrent Neural Networks Tutorial, Part 3 – Backpropagation Through Time and Vanishing Gradients." *WildML*. 2015. Web. http://www.wildml.com/2015/10/recurrent-neural-networks-tutorial-part-3-backpropagation-through-time-and-vanishing-gradients]

[7] "Mean Absolute Error." *Wikipedia*. Wikimedia Foundation. Web.

[8] "Linear Regression." *Wikipedia*. Wikimedia Foundation. Web.