Name: Prerana Chakraborty

Email address: prerana.chaks@gmail.com

Contact number: 9433056027

Anydesk address: **338 565 775**

Years of Work Experience: **0**

Date: **8th Dec 2020**

Self Case Study -1: Web Traffic Time Series Forecasting - Forecast future traffic to Wikipedia pages

---

"After you have completed the document, please submit it in the classroom in the pdf format."

Please check this video before you get started:
https://www.youtube.com/watch?time_continue=1&v=LBGU1_JO3kg

Overview

*** Write an overview of the case study that you are working on. ***(MINIMUM 200 words)*** ***

Introduction

This is the first self case study of Applied AI Course . The problem statement is **Web Traffic Time Series Forecasting - Forecast future traffic to Wikipedia pages** . Wikipedia pages are popular webpages which can sometimes see a hike in web traffic . So it is important that we know the traffic levels in advance in order to cope with the situation .

Business Problem

1. **Web traffic** is the amount of data sent and received by visitors to a website. This is generally determined by the number of visitors on the page .Sites monitor the incoming and outgoing traffic to see pages of their site are popular and if there are any apparent trends, such as one

specific page being viewed mostly by people in a particular country. Nowadays, web traffic forecasting is a major problem as this can cause setbacks to the workings of major websites.

2. Most of the people may have encountered a crashed site or very slow loading time for a website when there are a lot of people using it. This significantly affects user experience and they might leave a bad review for that site.So it is desirable that the owner of the site puts in place a traffic management plan to handle heavy traffic . This is where forecasting is needed .

ML Formulation

Data Overview

Data Source: https://www.kaggle.com/c/web-traffic-time-series-forecasting/data

There is a total of 6 data files .

- train_* - Training data which contains Pages as a column and the page hit values for a certain date range .
- key_* - Key files contains the short code for each page.
- Sample_submission _* - submission format

This problem was a part of a Kaggle competition . There were 2 stages for the competition .

First stage has data ranging from July, 1st, 2015 to December 31st, 2016. The second stage has training data till September 1st, 2017.

train_* has columns Page and the remaining are dates corresponding to the stages . The Page column has the page names and the date columns have the pagehit value that date.

Mapping real world to ML problem

This is a time series forecasting problem where I will be forecasting page hits of future dates from past dates.As I am forecasting numerical values so this is a regression type predictive modelling problem.

Business Objectives

- Identifying the nature of the phenomenon by establishing a pattern
- Forcasting future values.
- No low latency requirements but shouldn't take days.

Performance Metric

The metric that I will be using for this problem is Symmetric mean absolute percentage error(SMAPE) as it gives weightage to both positive and negative errors.

$$\text{SMAPE} = \frac{100\%}{n} \sum_{t=1}^{n} \frac{|F_t - A_t|}{(|A_t| + |F_t|)/2}$$

Research-Papers/Solutions/Architectures/Kernels

*** Mention the urls of existing research-papers/solutions/kernels on your problem statement and in your own words write a detailed summary for each one of them. If needed you can include images or explain with your own diagrams. it is mandatory to write a brief description about that paper. Without understanding of the resource please don't mention it***

1. https://www.kaggle.com/screech/ensemble-of-arima-and-lstm-model-for-wiki-pages

   This Kaggle Solution is an extension of the existing kernel.

   https://www.kaggle.com/muonneutrino/wikipedia-traffic-data-exploration

   Summary:

   - Loading and reading the train data into train_df
   - Exploratory Data Analysis
     - Average pageview for each language
       - The language of each of the page is extracted from the page name .
       - A dictionary lang_sets is created to store the train data corresponding to each language as the key.
       - Then the average pageview value for each date corresponding to each language is stored in a dictionary called sum .
       - Plot the sum value for each language .
       - From the plot I understood that the average pageview for English Language is high.
     - Autocorrelation and Partial Autocorrelation Plots
       - The acf and pacf is plotted for each language.

        - From the acf plots we find that there is a weekly trend for most of the languages .
        - From the pacf plots , the AR value is calculated depending on the lags that are outside the confidence intervals.
- Modelling - 2 models are prepared - ARIMA and LSTM
    - ARIMA
        - A model for each language is built considering the p,d,q values from the acf and pacf plots.
        - Model is trained on the average value for each language that is stored in the sums dictionary.
        - It is mentioned that the predictions from this model will be used as a input to the ensemble model.
    - LSTM
        - Vanilla LSTM of one layer with 8 neurons are used .
        - Optimizer used is RMSProp and loss metric is MSE
        - The first 549 days value is used to predict the next 549 days value . For example: Data from 1.7.2015 to 30.12.2016 is used for forecasting the values from 2.7.2015 to 31.12.2016.

2. https://github.com/Arturus/kaggle-web-traffic/blob/master/how_it_works.md

   I just went through the feature engineering and feature processing part of this document.

   Features generated are Country , agent and site name from the page name .Also year to year and quarter to quarter autocorrelation values are generated . Another feature called the page popularity is generated as the median value of the pageviews for each page .

   The pageview values are converted to log1p values . The country, agent and site name are one hot encoded. All the features are normalized . The pageview values are normalized individually for each page .

3. https://www.itm-conferences.org/articles/itmconf/pdf/2020/02/itmconf_icacc2020_03017.pdf

   In this paper , a new method for forecasting has been developed combining ARIMA and LSTM. Methodology steps discussed in this paper are the following :

   - Loading the time series data into an input vector .
   - Decomposing the vector into two parts by applying single level Discrete Wavelet Transform . This decomposition provides two data components , one is Approximate (A) component and another is Detailed(D) component.
   - ARIMA model is applied on the Detailed component and LSTM model is applied on the Approximate component to get 2 separate forecasts.

- Then the 2 forecasts are combined together using inverse DWT to get back the final forecast .

4. https://www.youtube.com/watch?v=y7KLbd7n75g

   It's a video explaining wavelets and wavelet transform . Wavelets allow us to have an idea about frequency magnitudes in a given time interval. In case of wavelet transform , a signal is broken down into set of mutually orthogonal wavelet basis functions.There is a mother wavelet $\psi(t)$. Smaller wavelets are derived from the parent wavelets

   $\psi_{a,b}(t) = (1/\sqrt{a}) * \psi((t-b)/a)$

   Where a is the scale and b is the position .

5. https://mil.ufl.edu/nechyba/www/eel6562/course_materials/t5.wavelets/intro_dwt.pdf

   This paper explains about Single level 1D DWT decomposition of a signal .

   The input vector x should be of length 2^n. The output vector w should be of same length.Let the input vector be of length 16.Now x is filtered with some discrete low pass filter h at intervals of 2 and the resulting vector is stored in the first 8 elements.of w .Then x is filtered with some discrete high pass filter at intervals of 2 and the resulting vector is stored in the last 8 elements of w .

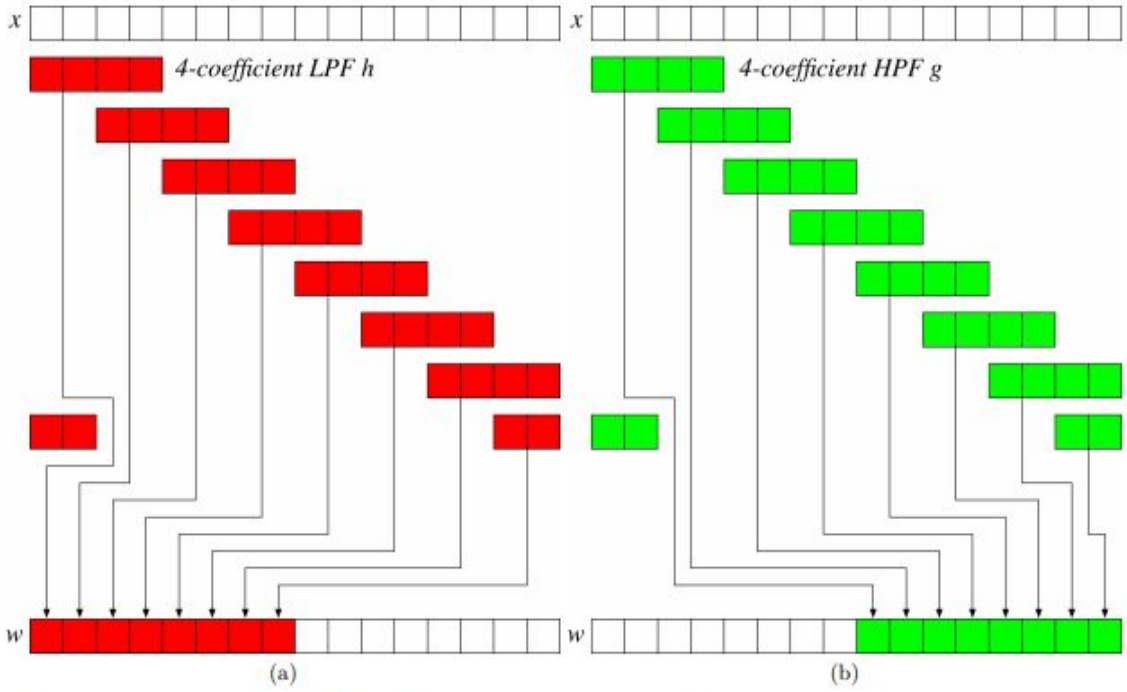


Figure 3: (a) First step of the DWT for a signal of length 16: The original signal is low-pass filtered in increments of two, and the resulting coefficients are grouped as the first eight elements of the vector. (b) Second step of the DWT: The original signal is high-pass filtered in increments of two, and the resulting coefficients are grouped as the last eight elements of the vector.

6. https://www.sciencedirect.com/topics/engineering/detail-coefficient#:~:text=Coefficients%20(weights)%20associated%20with%20the,%2C%20capture%20high%2Dfrequency%20information.

   Big Data and Monitoring The Grid

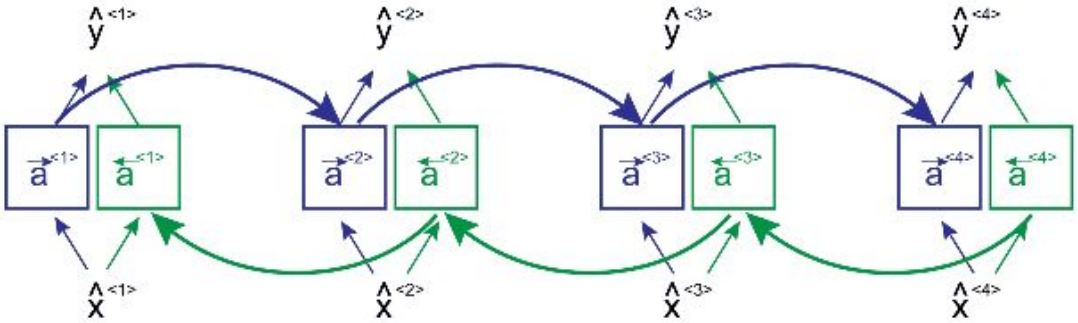
   Wavelet Transform

   Wavelet Transform functions are orthogonal to each other .Coefficients that are associated with low pass filter used for decomposition are called Approximation components and those associated with high pass filter are called Detailed components.The low pass filter captures low frequency information and high pass filter captures high frequency information.

7. https://valiancesolutions.com/bi-lstm-explained/

   Bidirectional LSTM is the result of clubbing 2 LSTM network - one in forward direction and the other in backward direction.I wish to use this type of network because sometimes the traffic is affected by future and past holidays. For example , if there is an upcoming holiday ,

then there might be an increase in the traffic. Again after a spike there is generally a fall in traffic, so to get an idea of both past and future , I want to use Bidirectional LSTM.



Connecting the backward cells

First Cut Approach

*** Explain in steps about how you want to approach this problem and the initial experiments that you want to do. ***(MINIMUM 200 words)*** ***

*** When you are doing the basic EDA and building the First Cut Approach you should not refer any blogs or papers ***

- First I want to find the average pageviews for each date across all the pages.This average pageview values will be used in EDA rather than plotting for all pages.
- EDA
    - Plotting the average pageviews vs. weekdays to get an idea how the pageviews change with weekdays .If there is a pattern for different weekdays then it will be used as a feature.
    - Plotting the average pageviews vs days coloring it based on weekdays and weekends to understand if there is any difference that if the traffic is high in weekends than in weekdays or vice versa . If there is a difference then weekends will also be included as a feature.
    - Plotting the average pageviews vs. months to get an idea how the pageviews change with months .Similarly if there is any pattern it will be added as a feature.

    - Finding the language of each page and plotting the distribution of the languages as a histogram.
    - Finding the agent of each page and plotting the distribution of the agent type as a histogram.
    - Finding the access type of each page and plotting the distribution of the access type as a histogram.
    - Plotting the average pageviews vs agent and average pageviews vs. access to see if there is a pattern . If there is a pattern they will be included as features.
    - Finding the pageview percent for each page and plotting the histogram of average pageview percent for each language.
    - Plotting the acf plots of the top 5 pages based on their view percentage. To get the timestep value for LSTM.
- Feature Development
    - Depending on the EDA , the features will be decided and only those which shows some pattern will be included.
    - Target Feature will be the pageview value for the dates .
    - All the features will be Standardized .
- Model Development
    - LSTM Model
        - Bidirectional LSTM Model will be used to capture both past and future trends.
        - I want to use timestep as 7 as I have seen in the reference kernel that there is a weekly trend.
        - I want to train the model on a page that has a high view percentage .
        - Train data will have the first 70% dates white test data will have the next 30% dates.
        - I want to use SMAPE as the metric for evaluation .

Notes when you build your final notebook:

1. You should not train any model either it can be a ML model or DL model or Countvectorizer or even simple StandardScalar
2. You should not read train data files

3. The function1 takes only one argument “X” (a single data points i.e 1*d feature) and the inside the function you will preprocess data point similar to the process you did while you featurize your train data
    a. Ex: consider you are doing taxi demand prediction case study (problem definition: given a time and location predict the number of pickups that can happen)
    b. so in your final notebook, you need to pass only those two values
    c. def final(X):
        preprocess data i.e data cleaning, filling missing values etc
        compute features based on this X
        use pre trained model
        return predicted outputs
    final([time, location])
    d. in the instructions, we have mentioned two functions one with original values and one without it
    e. final([time, location]) # in this function you need to return the predictions, no need to compute the metric
    f. final(set of [time, location] values, corresponding Y values) # when you pass the Y values, we can compute the error metric(Y, y_predict)
4. After you have preprocessed the data point you will featurize it, with the help of trained vectorizers or methods you have followed for your train data
5. Assume this function is like you are productionizing the best model you have built, you need to measure the time for predicting and report the time. Make sure you keep the time as low as possible
6. Check this live session: https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/4148/hands-on-live-session-deploy-an-ml-model-using-apis-on-aws/5/module-5-feature-engineering-productionization-and-deployment-of-ml-models