Wikipedia Web Traffic - Time Series Analysis and Prediction

Bhumika Chopra (2018MT10748), Hetvi Jethwani (2018MT10754), Sharut Gupta (2017MT60250), Silky Singh (2018MT10769), and Utkarsh Gupta (2017MT10753)

Department of Mathematics, Indian Institute of Technology, Hauz Khas, New Delhi, India

ABSTRACT

Traffic prediction of web-pages is an important problem, and applications in areas like marketing have increased the need for effective prediction of values of multiple time series. We apply an array of statistical techniques over the "Web Traffic Time Series Forecasting" data-set to predict future traffic of Wikipedia articles, and use various metrics like MSE, SMAPE, etc to survey these models. We support our modelling with a thorough exploratory analysis of the dataset, and draw various insights about the given data. With our comprehensive analysis, we conclude that LSTM-based models perform the best.

Keywords: Web-traffic, ARIMA, Forecasting, LSTM, L2 Regression, SMAPE, prediction

1 INTRODUCTION

Wikipedia is free multilingual internet-based encyclopedia (Wikipedia contributors, 2021) which is maintained by a community of volunteers. The encyclopedia is open-collaborative that is readers can contribute articles, correct errors, provide citations and dispute facts. Wikipedia is the world's largest encyclopedia and the most accessible compilation of knowledge to have ever existed in human history. Wikipedia has often cited as the greatest open-source production model that has revolutionized democratization of knowledge.

Web traffic forecasting and analysis has many applications for both the users and the website owners. Popularity, trend and patterns in traffics on particular webpages help website owners design more robust security protocols against issues such as Denial-of-service attacks, structure the content of webpages to improve content-consumption experience for the users, optimize storage and network resources, and understand user consumption patterns to make better business decisions.

Forecasting web-traffic is an active area of research. The authors in (Petluri and Al-Masri, 2018) developed a consistent forecasting model to predict the future traffic of Wikipedia pages. Another research article (Yao et al., 2006) proposed wavelet pattern analysis and neural networks for predicting web traffic which used highly sophisticated algorithms to predict web-traffic with increased accuracy. But there approach was marred by there time-intensive algorithms and therefore was impractical for real-time analysis. Other approaches have included using genetic algorithms and neural networks (Chen, 2011) and Generative Adversarial Models (Zhou et al., 2021). Research in the area has been increasing steadily over the years and new models are being used and optimized everyday to solve the Web-Traffic forecasting models. Recently, (Casado-Vara et al., 2021) proposed using a variant of LSTM models for the same.

For the "Web Traffic Time Series Forecasting" dataset, we first detail the exploratory data analysis- and infer statistical patterns from various features that the dataset has, for eg. page-specific patterns, periodicity, etc. Then, we proceed to predict timeseries using the ARIMA Model, Linear Regression, L2 Regression, and Statistical Machine Learning Techniques like using LSTMs, Random Forest, ADABoost, etc.

1.1 Data Collection and Sources

The dataset used for our term paper is "Web Traffic Time Series Forecasting dataset" provided by Google. The dataset has time series with respect to Wikipedia articles. Every time-series represents views per day from 1st July, 2010 to 31st December, 2016 for 145,000 Wikipedia articles in 7 different languages, and

additional pages with multimedia. We are also given the source of traffic represented by the corresponding time series (i.e. is it from all devices, was it from a desktop browser, was it from mobile, or was it by a web crawler). We don't distinguish between sources of traffic in our discussion- we consider traffic from all devices.

2 EXPLORATORY DATA ANALYSIS

2.1 Data description

In the following section, we use plots to describe and represent our time-series data, this will help in better visualization. We can see that most data is available for English (which makes sense, since it is also the most widely spoken language). In devices used for access, we see that count of accesses via desktop and mobile is roughly the same, so the popularity and widespread usage of mobile phones can be inferred from this.

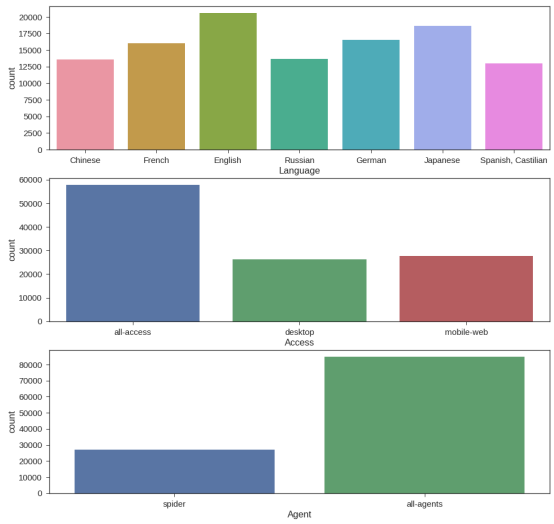


Figure 1. Variation using bar-charts

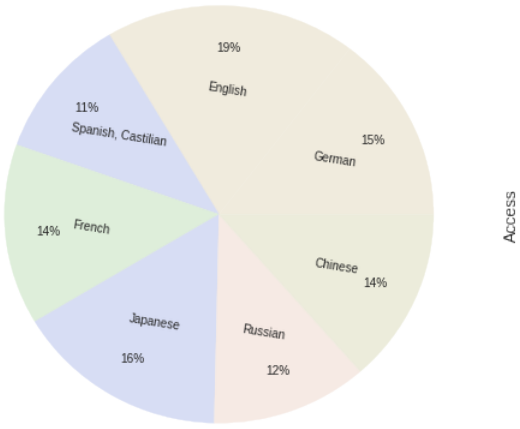


Figure 2. Showing proportions of types of pages

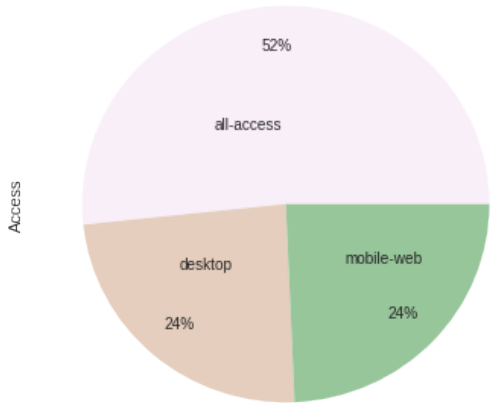


Figure 3. Access

We plot the density plots of mean and maximum views for pages across languages. We observe that

- All languages follow similar patterns. They are bimodal with with peakes around 10-30 and 200-300 views in the mean view. The second peak surprisingly is has a higher amplitude.
- The English pages (mustard), that is pages have the highest Mean and Average Views which is to be expected (most accessible language in countries with easy internet access)
- The Chinese pages (pink) have lower mean and max views. This may be attributed to high censorship of China.

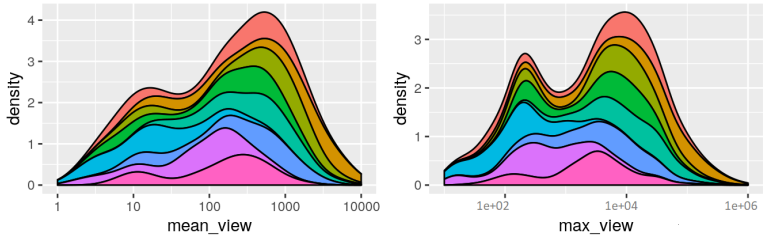


Figure 4. Density of Mean and Max views

We further give the bar plots (with error bars) and box plots of different languages across access-types. These illustrate the differences across languages

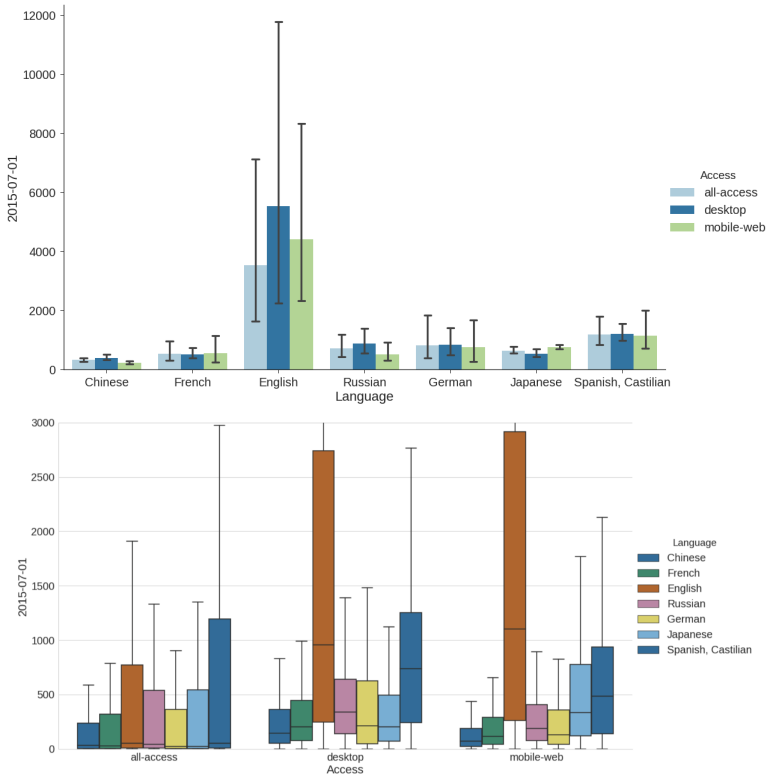


Figure 5. Density of Mean and Max views

2.2 Languages and Traffic

In the following section we discuss effects of different languages on web traffic. As we can see in figure 1, the traffic for English language pages is consistently the highest- which is plausible since English is the most widely spoken language in the world. We also observe a spike in traffic around August 2016 and November 2016. It is interesting to see a spike in the Russian language around that time. A possible

reason for this spike is the elections in the USA, in which there were rumours of Russian involvement throughout.

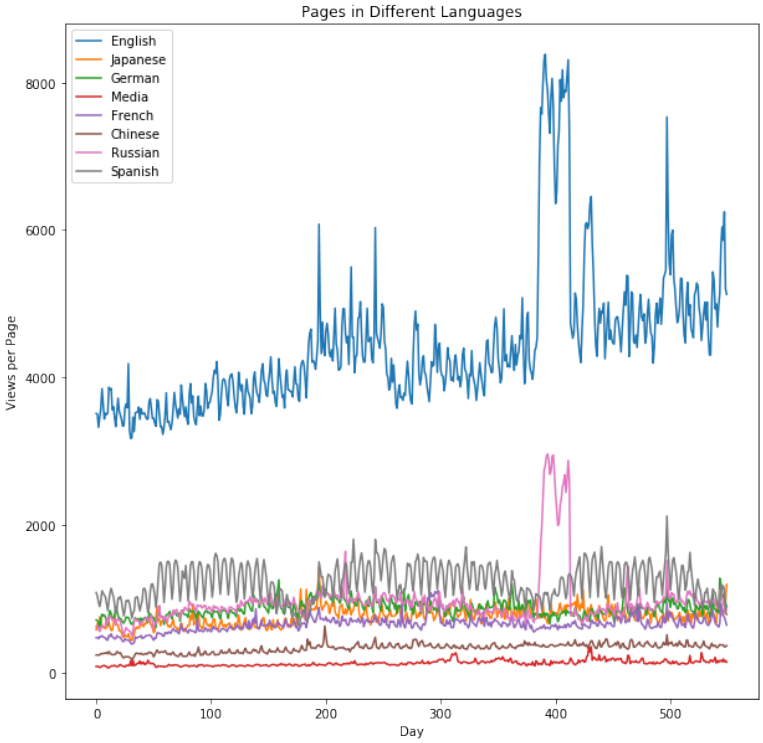


Figure 6. Change of views/day over time

We can also observe differences in periodic nature of traffic from Fig. 2. For example, the Spanish language data seems to show the strongest periodic nature, as opposed to the Russian which is mainly low but peaks around August 2016. Thus, from this section we conclude that language plays an important factor in traffic, and should be accounted for while predicting forecasts. Autocorrelation is a statistical tool for analysis of periodic patterns in data. From this plot, we can observe a weekly trend in traffic of Spanish language webpages- which indicates some variation in browsing habits on weekdays and weekends. Periodicity of traffic may be affected due to major events happening in the world. Eg. it is reasonable to expect a spike in searches related to FIFA World Cup every 4 years.

2.3 Periodicity in Spanish

In the following section we infer that traffic patterns for Spanish webpages are periodic, and we discuss their periodicity.

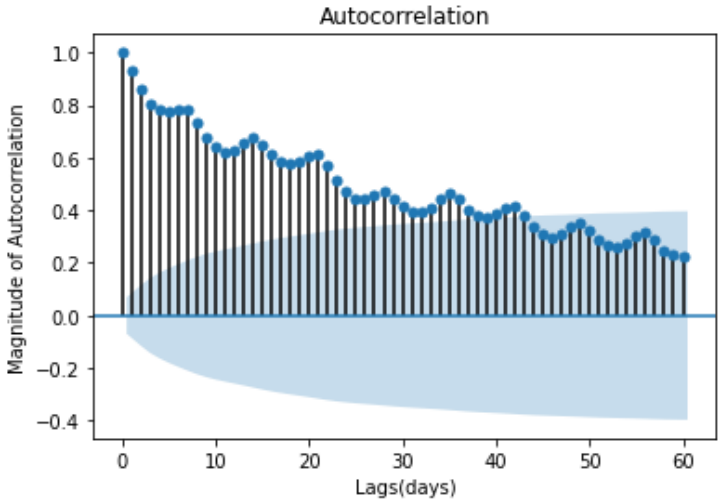


Figure 7. Autocorrelation of Spanish traffic

2.4 Timeseries of individual Wiki pages

This can help us gain insights on traffic patterns and how they relate to everyday news. For example, traffic at "Hunky Dory", one of David Bowie's most famous albums, peaked around his death in January 2016 (Fig. 3). On the contrary, we see that the Internet of Things wiki page became gradually mainstream as traffic slowly increased in 2016 (Fig. 4) and there was no singular event which caused a transient peak.

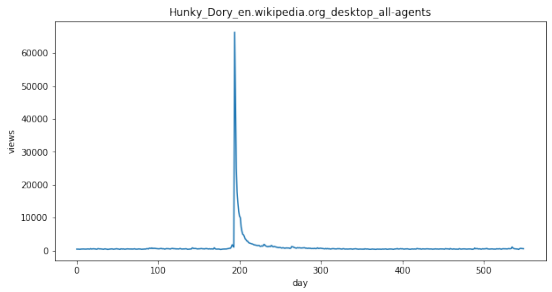


Figure 8. Peak in traffic at "Hunky Dory" in 2016

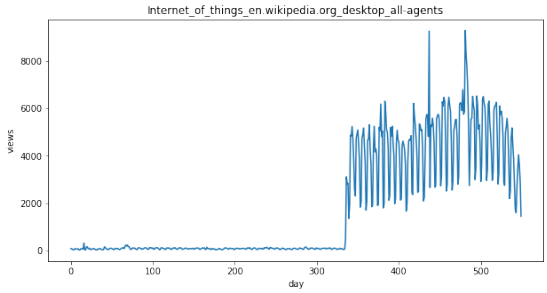


Figure 9. Gradual increase in traffic for IoT

3 METHODS

3.1 Feature engineering and preprocessing

We extracted the following features from the dataset:

- *pageviews* (spelled as 'hits' in the model code, because of my web-analytics background). Raw values transformed by log1p() to get more-or-less normal intra-series values distribution, instead of skewed one.
- *Agent, country, site* - these features are extracted from page urls and one-hot encoded
- *day of week* - to capture weekly seasonality
- *year-to-year autocorrelation, quarter-to-quarter autocorrelation* - to capture yearly and quarterly seasonality strength.
- *page popularity* - High traffic and low traffic pages have different traffic change patterns, this feature (median of pageviews) helps to capture traffic scale. This scale information is lost in a pageviews feature, because each pageviews series independently normalized to zero mean and unit variance.

3.2 ARIMA Model

An ARIMA model is a class of statistical models for analyzing and forecasting time series data. ARIMA can capture complex relationships as it takes error terms and observations of lagged terms. These models rely on regressing a variable on past values. The model takes 3 parameters $p, q, and\, d$ as input.

As the name suggests the model is

- **Auto Regressive-** the model applies a weight to each of the past variable in the time series and the weights can vary based on how recent they are to predict future values of the variables. Characterized by parameter p
- **Integrated-** the model reduces seasonality from a time series and makes it stationary. Characterized by parameter d
- **Moving Average-** the model removes non-determinism or random movements from a time series. Characterized by parameter q.

We used the ADF (augmented Dickey–Fuller) test and found the time series to be non-stationary. After the first difference (d=1), the differenced series becomes stationary. Now, this differenced series is considered.

3.3 Linear Regression

Linear regression (LR) is a commonly used type of predictive analysis. It attempts to model the relationship between two variables by fitting a linear equation to observed data. The target variable is the dependent variable while the explanatory variable is the dependent variable. For this task of web traffic forecasting, we define two metrics:

1. Mean Absolute Percentage Error (MAPE): It is a measure of prediction accuracy based on absolute errors and is used as a substitute of loss function in linear regression. Mathematically, it can be expressed as

$$MAPE = \frac{100}{n} \sum_{i=1}^{n} \frac{A_i - \hat{F}_i}{A_i}$$

2. Symmetric Mean Absolute Percentage Error (SMAPE): SMAPE is an accuracy measure based on the relative error value. Mathematically, it can be expressed as

$$SMAPE = \frac{100\%}{n} \sum_{i=1}^{n} \frac{2(F_i - A_i)}{|A_i| + |F_i|}$$

where A_t is the actual value and F_t is the forecast value.

Figure 11a show the plots for the modelled and fact values along with prediction intervals and anomalies. As shown, the LR model achieved a MAPE score of 18.18% and a SMAPE score of 18.81%. In order to improve the model's performance, additional feature engineering was done to extract relevant features from the dataset. This includes features like if the given day was weekday or weekend etc. As observed from Figure 11b, The MAPE and SMAPE errors decreased a bit, from 18.18% and 18.81% to 17.30% and 17.90%. When plotted the coefficients of the LR model, it was found that "is weekend" feature showed up as useful resource, while the "weekday" feature worsened the performance.

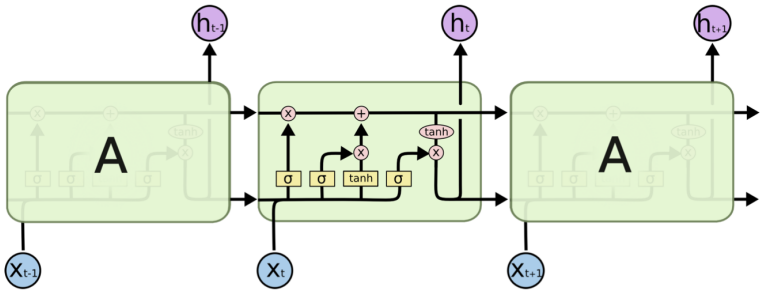
3.4 Statistical Machine learning Techniques

In this subsection, we tested different statistical machine learning techniques like ADABoost, Gradient Boost and Random Forests. Predicted vs ground truth values were analyzed based on the RMSE value for each of the models. The best results obtained were as follows:

- **Adaboost Regressor:** With Decision Tree Regressor of max depth 4 as base estimator, n estimators as 5000 and learning rate as 0.01 we achieved the highest performance.
- **Gradient Boost Regressor:** Best performance for this model we obtained at a learning rate of 0.1 and max depth of 4
- **Random Forrest Regressor:** The best performing hyperparameters for this model include number of estimator as 500 and maximum depth of 4.

3.5 LSTM

Long Short Term Memory networks – usually just called LSTMs, are a special kind of RNN, capable of learning long-term dependencies. They were introduced by Hochreiter and Schmidhuber (1997), and were refined and popularized by many people in following work.1 They work tremendously well on a large variety of problems, and are now widely used. LSTMs are explicitly designed to avoid the long-term dependency problem. Remembering information for long periods of time is practically their default behavior, not something they struggle to learn. LSTMs have a chain like structure, but the repeating module is a system of four neural network layer, interacting in a very special way. The Long Short-Term



The repeating module in an LSTM contains four interacting layers.

Memory network, or LSTM for short, is a type of recurrent neural network that achieves state-of-the-art results on challenging prediction problems. They provide an elegant solution to sequence prediction problems. For this experiment, Keras library of Python has been used.

3.6 Prophet

Prophet works as an additive regression model which decomposes a time series into (i) a linear/logistic trend, (ii) a yearly seasonal component, (iii) a weekly seasonal component, and (iv) an optional list of important days (such as holidays, special events).

It claims to be "robust to missing data, shifts in the trend, and large outliers", which would make it well suited for this particular task. It is implemented in an R library, and also a Python package. Prophet also offers a decomposition plot, where we can inspect the additive components of the model: trend, yearly seasonality, and weekly cycles.

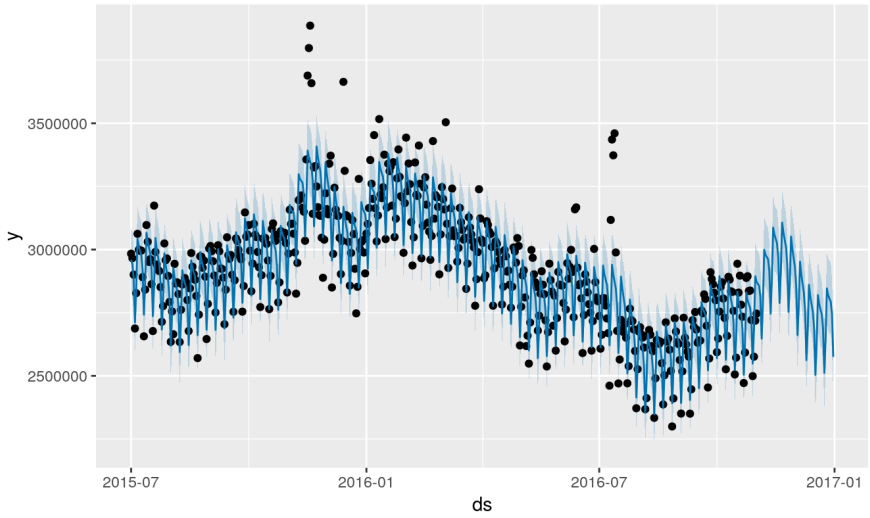


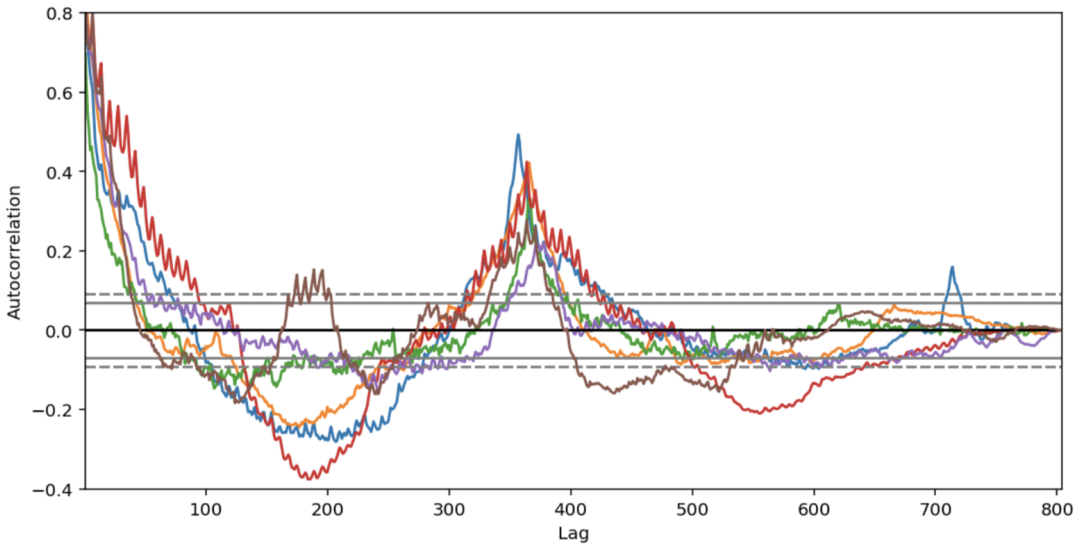
Figure 10. Standard prophet forecast model

The observed data are plotted as black points and the fitted model, plus forecast, as a blue line. In light blue we see the corresponding uncertainties.

3.7 Key Prediction Information

There are two main information sources for prediction:

- **Local features:** If we see a trend, we expect that it will continue (AutoRegressive model), if we see a traffic spike, it will gradually decay (Moving Average model), if wee see more traffic on holidays, we expect to have more traffic on holidays in the future (seasonal model).
- **Global features:** If we look at autocorrelation plot in Figure **??**, we'll notice strong year-to-year autocorrelation and some quarter-to-quarter autocorrelation. The good model should use both global and local features, combining them in a intelligent way.



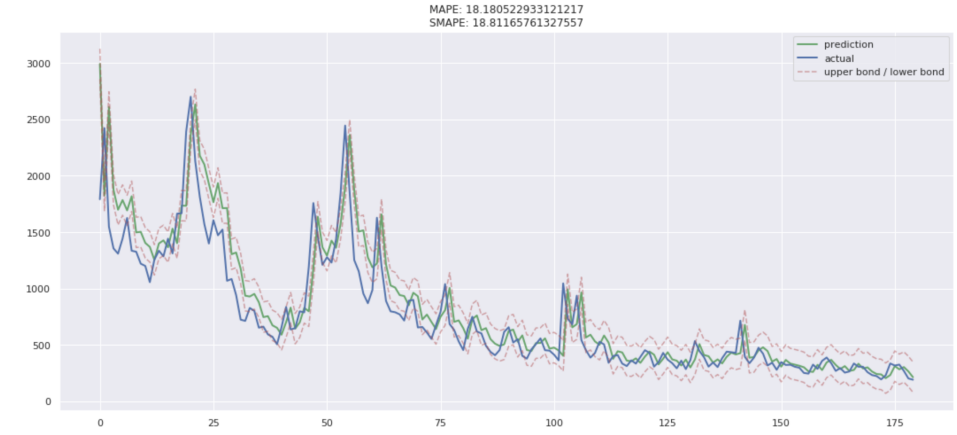
4 RESULTS

4.1 Linear Regression

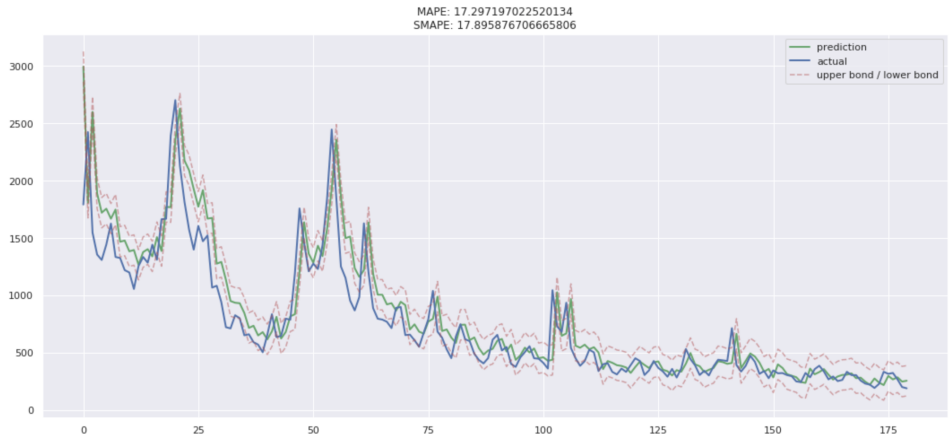
4.2 Hypothesis Generation and Validation

In Hypothesis Generation, we first list out all the possible factors that can affect the outcome. Then we test each hypothesis to test the validity of every such possibility. Hypothesis generation is done before having a look at the data in order to avoid any bias that may result after the observation. It helps us to point out the factors which might affect our dependent variable.
As discussed above, we list down the hypotheses which might affect the traffic on wiki pages:

- **Claim 1: There will be an increase in the traffic as the years pass by**
  Population has a general upward trend with time, so we can expect more people to access the pages. Also, with the advancement of technology and access to more population, the traffic is expected to increase. As observed from Figure 12(a) and (b), Exponential growth is noticed year by year which validate our first hypothesis.
- **Claim 2: Traffic on weekdays will be more as compared to weekends/holidays**
  People usually go to offices on weekdays and hence the traffic is expected to be more. It can be inferred from Figure 12(c) and (d), that the traffic is more during the weekdays as compared to weekend. Also note that the peak appears to be around Tuesday and is monotonically decreasing as the weekend approaches.
- **Claim 3: Traffic during the peak hours will be high**
  People will travel to work or college and hence are expected to access internet during morning or lunch. Similarly, youngsters are expected to watch entertainment shows around evening or



(a)



(b)

Figure 11. Plots of modelled and fact values along with prediction intervals and anomalies using (a) Vanilla Linear Regression (b) Linear Regression with additional features

night. From Figure 12(e), It can be observed that the peak traffic in the evening is at 7 PM. Then a decreasing trend is noticed till 5 AM. After that the passenger count starts increasing again and peaks again between 11AM and 12 Noon.

4.3 Hypothesis Testing

We test the claims made in the previous subsection.

- Testing claim 1: **There will be an increase in the traffic as the years pass by**
  To test this hypothesis, we compute and compare mean views of days 1 year apart. Refer to table **??** for results of this testing claim. We run the following tests for the same:

  – One-way Anova Test
  The one-way analysis of variance (ANOVA) Test is used to determine whether there are any statistically significant differences between the means of three or more independent (unrelated) groups. Here since we compare the means for each day of the year, we have 365 different days.

  –

- Testing claim 2: **There will be more traffic on weekdays as compared to weekends**

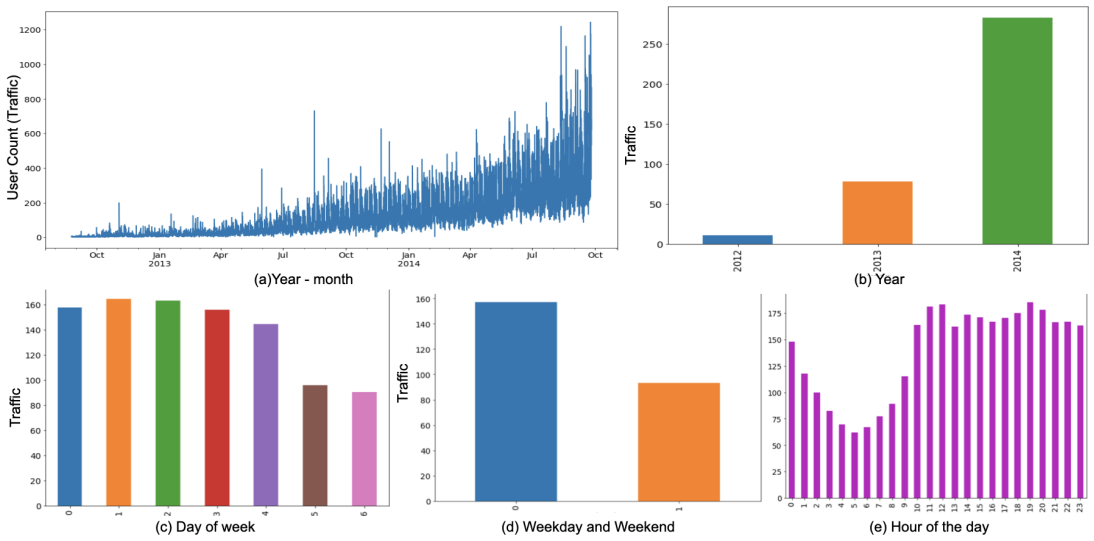


Figure 12. (a) Represents a time series of traffic versus month-year; (b) Represents a bar plot of traffic versus year; (c) Indicates traffic on working days of the week (d) Shows a bar plot of traffic on weedays versus weekends; (e) A bar plot indicating traffic at different hours of the day.

μ_0 : Mean on corresponding day μ_1 : Mean a year later on the same day				
H_0 : Mean views of days 1 year apart are the same, i.e. $\mu_0 = \mu_1$				
Test ↓	Day →	Mon	Thu	Sun
One-way ANOVA	Statistic:	0.45	0.81	0.95
	p value:	0.50	0.36	0.32
	Reject Null:	No	No	No
Median Test	Statistic:	2267.7	2258.9	2345.1
	p value:	0.0	0.0	0.0
	Grand median:	89.0	85.0	90.0
	Reject Null:	Yes	Yes	Yes
One sided T test $H_1 : \mu_0 < \mu_1$ equal variance false	Statistic:	-0.67	-0.90	-0.97
	p value:	0.25	0.18	0.16
	Reject Null:	No	No	No
One sided T test $H_1 : \mu_0 < \mu_1$ equal variance true	Statistic:	-0.67	-0.90	-0.97
	p value:	0.25	0.18	0.16
	Reject Null:	No	No	No

Comparing views on a corresponding day, 1/2 a year later, and 1 year later on same day				
Test ↓	Day →	Mon	Thu	Sun
One-way ANOVA	Statistic:	0.52	0.47	1.01
	p value:	0.59	0.62	0.36
	Reject Null:	No	No	No
Median Test	Statistic:	2441.1	2292.2	2581.1
	p value:	0.0	0.0	0.0
	Grand median:	96.0	84.0	97.0
	Reject Null:	Yes	Yes	Yes

- Testing claim 3: **Distribution of traffic**
- Testing claim 4: **Comparision of views according to languages**

Comparing views on one weekday and weekend			
Test ↓	Day →	Mon [5-7-2015]	Sun [6-7-2015]
	Mean views:	1108.08	1044.34
One-way ANOVA	Statistic:	0.05	
	p value:	0.81	
	Reject Null:	No	
Median Test	Statistic:	0.0013	
	p value:	0.97	
	Grand median:	59.0	
	Reject Null:	No	
One-way T Alternate is mean on Mon >mean on Sun	Statistic:	0.239	
	p value:	0.40	
	Reject Null:	No	

Comparing views on all Mondays with on all Sundays			
Test ↓	Day →	Mon	Sun
	Mean views:	1386.45	1350.34
One-way ANOVA	Statistic:	0.97	
	p value:	0.32	
	Reject Null:	No	
Median Test	Statistic:	218.63	
	p value:	≈ 0	
	Grand median:	113.0	
	Reject Null:	Yes	
One-way T Alternate is mean on Mon >mean on Sun	Statistic:	0.98	
	p value:	0.16	
	Reject Null:	No	

Comparing views on all Mondays, Thursdays, and Sundays				
Test ↓	Day →	Mon	Thur	Sun
	Mean views:	1386.45	1285.66	1350.34
One-way ANOVA	Statistic:	4.12		
	p value:	0.016		
	Reject Null:	Yes		
Median Test	Statistic:	1703.56		
	p value:	≈ 0		
	Grand median:	111.0		
	Reject Null:	Yes		

4.4 Correlation

We use the correlation coefficient to study the strength of dependence between various extracted measure such as mean views per day, maximum views per day, variability in trends, etc. What we have observed from the plots is -

- There is a clear correlation between mean views and maximum views. Also here we find again the two cluster peaks we had identified in the individual histograms. A couple of outliers and outlier groups are noticeable.
- There is a clear trend toward lower viewing numbers on the weekend (Fri/Sat/Sun), and also a declining trend from Monday through Thursday. This gives us valuable information on the general type of variability over the course of a week.
- We find that articles with higher average view-count have more variability in their linear trends. We observe that the slopes of low-view articles are on average slightly higher than those of high-view articles.

Comparing aggregate weekday and weekend views			
Test ↓	Day →	Weekday	Weekend
	Mean views:	1313.92	1299.34
One-way ANOVA	Statistic:	0.49	
	p value:	0.47	
	Reject Null:	No	
Median Test	Statistic:	218.09	
	p value:	≈ 0	
	Grand median:	109.0	
	Reject Null:	Yes	
Two-way T-Test equal variance true	Statistic:	0.70	
	p value:	0.48	
	Reject Null:	No	
Two-way T-Test equal variance false	Statistic:	0.71	
	p value:	0.47	
	Reject Null:	No	

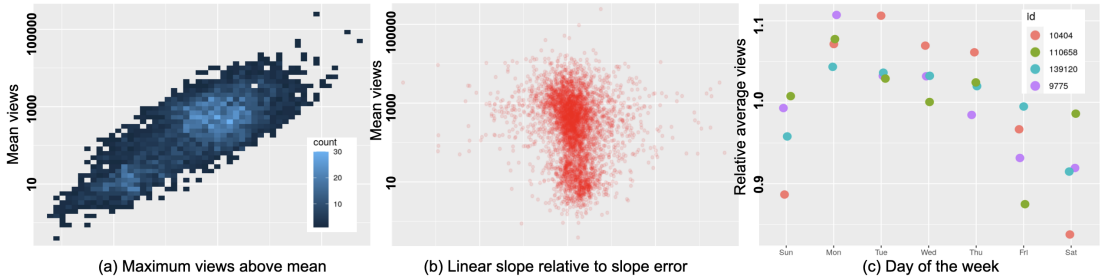
2-sided K-S Test Against Normal Distribution			
Day	Statistic	P-value	Reject Null
Mon	0.79	≈ 0	Yes
Thu	0.78	≈ 0	Yes
Sun	0.79	≈ 0	Yes

Comparing views language wise				
Test ↓	Day →	Japanese	Russian	French
	Mean views:	632.39	640.98	502.74
One-way ANOVA	Statistic:	0.77		
	p value:	0.46		
	Reject Null:	No		
Median Test	Statistic:	299.07		
	p value:	≈ 0		
	Grand median:	111.0		
	Reject Null:	Yes		

4.5 Generative Adversarial Network(GAN)

We present the Generative Adversarial Network(GAN) with Long-Short Term Memory(LSTM) as generator and deep Multi-Layer Perceptron(MLP) as discriminator to forecast the web traffic time series. The forecasting performances was compared with the traditional statistical methods and the deep generative adversarial network. Finally, we discuss a hybrid method to improve the overall accuracy of forecasting methods. We also found that time series with week cycle as decompose method was acceptable.

- **Naive approach**: In this approach, we simply take the last observed value as the forecast value.
- **Mean method:** In this method, we calculate the mean of the observed data as the web traffic forecast.
- **Exponential Smoothing (ETS)** We modeled the time series with ETS(A,N,N) i.e., simple exponential smoothing with additive errors.
- **ARIMA model**: The trial and error method was used to select ARIMA model parameters. The smaller AIC, AICc, and BIC, the better ARIMA model. The key parameters involved in the model; p is for AR, q for MA, and d is for difference orders. The classical ARIMA model is defined as



(a) Maximum views above mean (b) Linear slope relative to slope error (c) Day of the week

follows:

$$\left(1-\sum_{i=1}^{p}\alpha_i L^i\right)(1-L)^d X_t = \left(1+\sum_{i=1}^{q}\theta_i L^i\right)\varepsilon_t$$

where L is the lag operator, α_i is parameter of AR, θ_i is parameter of the MA, and ε_t is the parameter of the error term. AIC, AICc, and BIC metric for parameter set (2,1,1) is slightly below the (3,0,0) which suggests that ARIMA(2,1,1) is better than ARIMA(3,0,0).

We compare the ARIMA(2,1,1) and ETS(A,N,N) under the metrics of RMSE, MAE, MAPE and MASE and the results are given in the table below:

ETS(A,N,N)	RMSE	MAE	MAPE	MASE
Training Set	47.71	29.80	122.49	0.83
Test Set	10.04	10.04	12.39	0.28
ARIMA(2,1,1)	RMSE	MAE	MAPE	MASE
Training Set	48.53	30.29	122.01	0.84
Test Set	3.89	3.89	4.80	0.11

- **Hybrid method:** The combination is the mean summation of all four methods(ETS, ARIMA, NNAR and TBATS) divided by 5.

- **CNN-LSTM:** We used CNN-LSTM combined deep neural network as the G. Since the hyperparameters determine the goodness of the neural network and the univariate time series are relatively small, therefore grid search is used following these steps:
(I) Split the data into train and test sets.
(II) Use supervised learning to train (samples contains data points and forecasting).
(III) Moveforward validation, after each forecast is made one step forward, the true observation is added to the test dataset, allowing the model to predict using the most recent history. All predictions are compared to the true values and RMSE is calculated.
(IV) Repeat evaluation. Neural network model's stochasticity may result a different set of weights each time the model is trained which will in turn have a different performance. Repeated evaluation via move-forward and average error rate is adopted to counter this problem.
(V) Compare the performance results and select the best parameters.

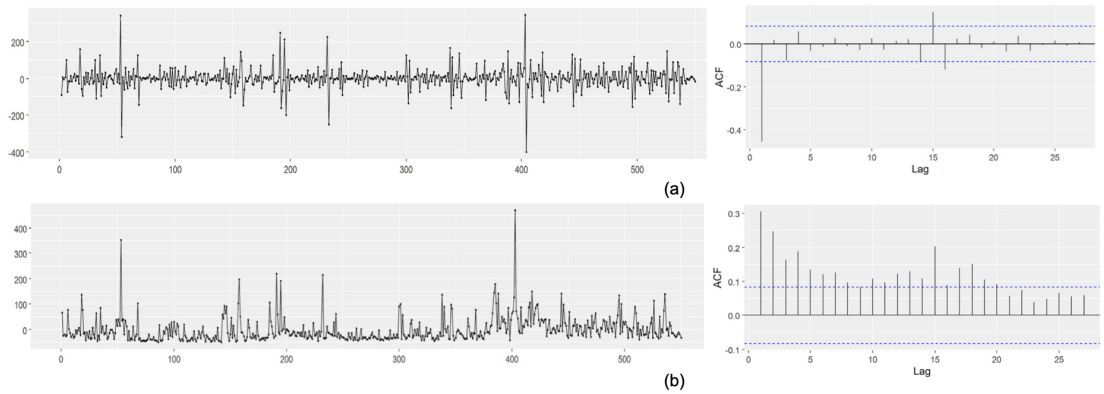


Figure 13. (a) Residuals and ACF for Naïve method; (b) Residuals and ACF for Mean method

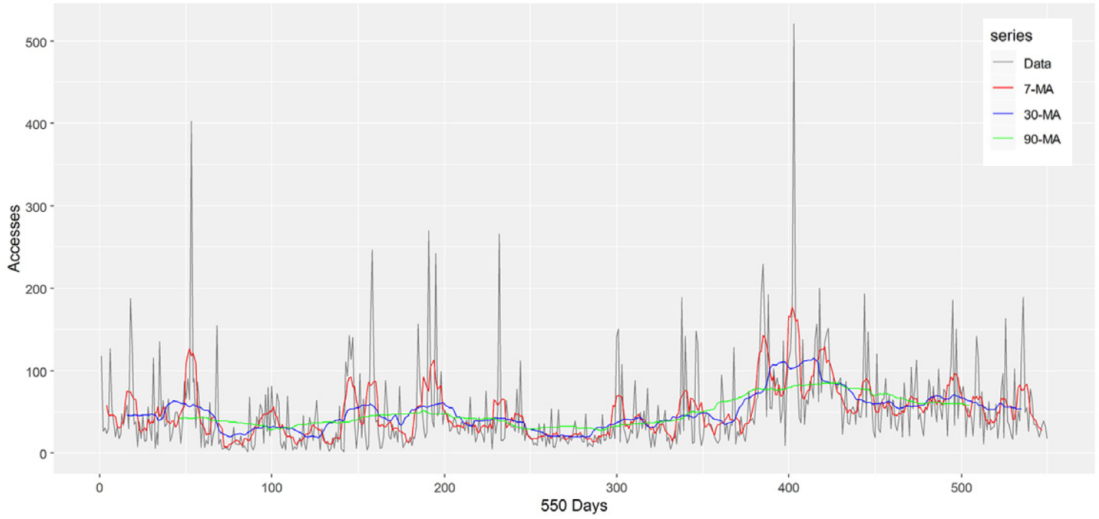


Figure 14. Moving Average plot for times series.

4.6 ARIMA Model

Parameters	MSE	AIC
(0, 0, 4)	MSE=34.201	AIC=5382.980
(0, 0, 5)	MSE=34.080	AIC=5382.054
(0, 0, 6)	MSE=34.497	AIC=5371.283
(0, 0, 7)	MSE=34.298	AIC=5373.053
(0, 1, 4)	MSE=88.249	AIC=5377.285
(0, 1, 5)	MSE=101.674	AIC=5379.17
(0, 1, 6)	MSE=78.972	AIC=5378.373
(0, 1, 7)	MSE=63.096	AIC=5367.896
(5, 0, 4)	MSE=45.165	AIC=5359.160
(5, 0, 5)	MSE=85.835	AIC=5343.798
(5, 0, 6)	MSE=68.060	AIC=5340.202
(5, 0, 7)	MSE=55.496	AIC=5318.691
(5, 1, 5)	MSE=141.144	AIC=5342.774
(5, 1, 6)	MSE=91.898	AIC=5323.920
(5, 1, 7)	MSE=99.281	AIC=5323.621

Table 1. ARIMA model with different Parameters

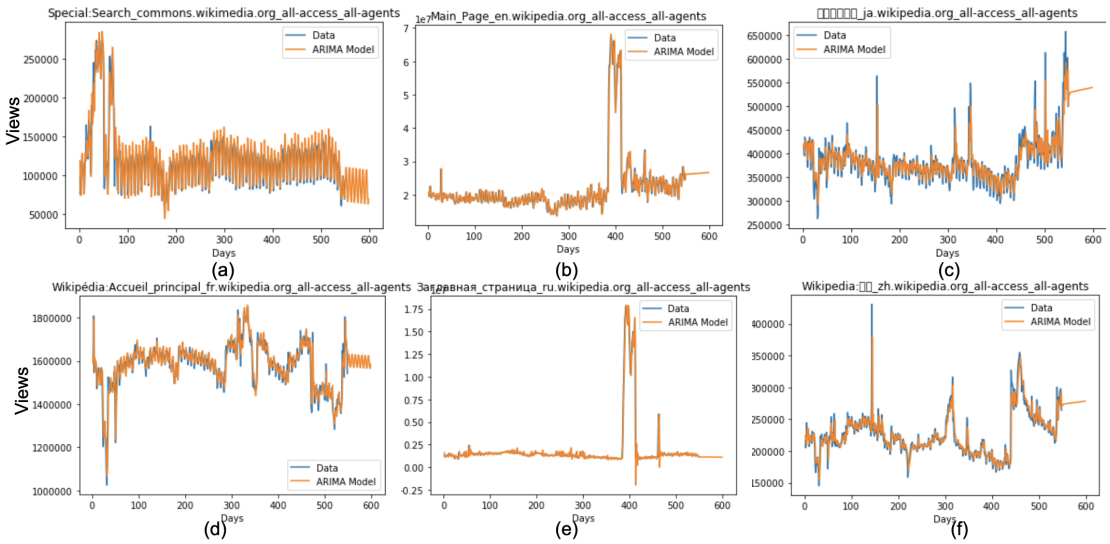


Figure 15. Time series analysis using ARIMA(2,1,2) on different sets of pages

We observe that in some cases the ARIMA model is able to predict the weekly substructure of the signal, which is good. In other cases it seems to just give a linear fit. Fitting the model to a subset of the data gave better results than fitting it over the complete data set.

4.7 Linear Regression

Further, two different types of regularization techniques were tried: Ridge and Lasso. These are used to reduce model complexity and prevent the over fitting of simple linear regression model. From Figure 24, we infer that despite strong regularization techniques, we do not see any significant improvement in the vanilla results.

4.8 L2 Regression

The purpose of this section is to show some simple transformations that allow out of box non-parametric L2 regressors to achieve better SMAPE scores. We can get these algorithms to get closer to optimizing SMAPE by changing the target variables by an invertible transformation. However a transformation alone won't make the algorithm actually optimize on SMAPE. Furthermore, fitting on transformed target variables will change a parametric model (e.g. linear regression), but a non-parametric model (e.g. decision tree or neural network) will be okay as it has no strict structure. SMAPE actuallr measures the average of $\frac{|p-t|}{|p|+|t|}$, p and t are scalars. By letting $r = \frac{p}{t}$, we get a single variable function in r.

For this purpose, pandas, numpy and sklearn have been used.

4.9 Statistical Machine learning Techniques

In this subsection, we tested different statistical machine learning techniques like ADABoost, Gradient Boost and Random Forests. Predicted vs ground truth values were analyzed based on the RMSE value for each of the models. The best results obtained were as follows:

- Adaboost Regressor: With Decision Tree Regressor of max depth 4 as base estimator, n estimators as 5000 and learning rate as 0.01 we achieved the highest performance.
- Gradient Boost Regressor: Best performance for this model ws obtained at a learning rate of 0.1 and max depth of 4
- Random Forrest Regressor: The best performing hyperparameters for this model include number of estimator as 500 and maximum depth of 4.

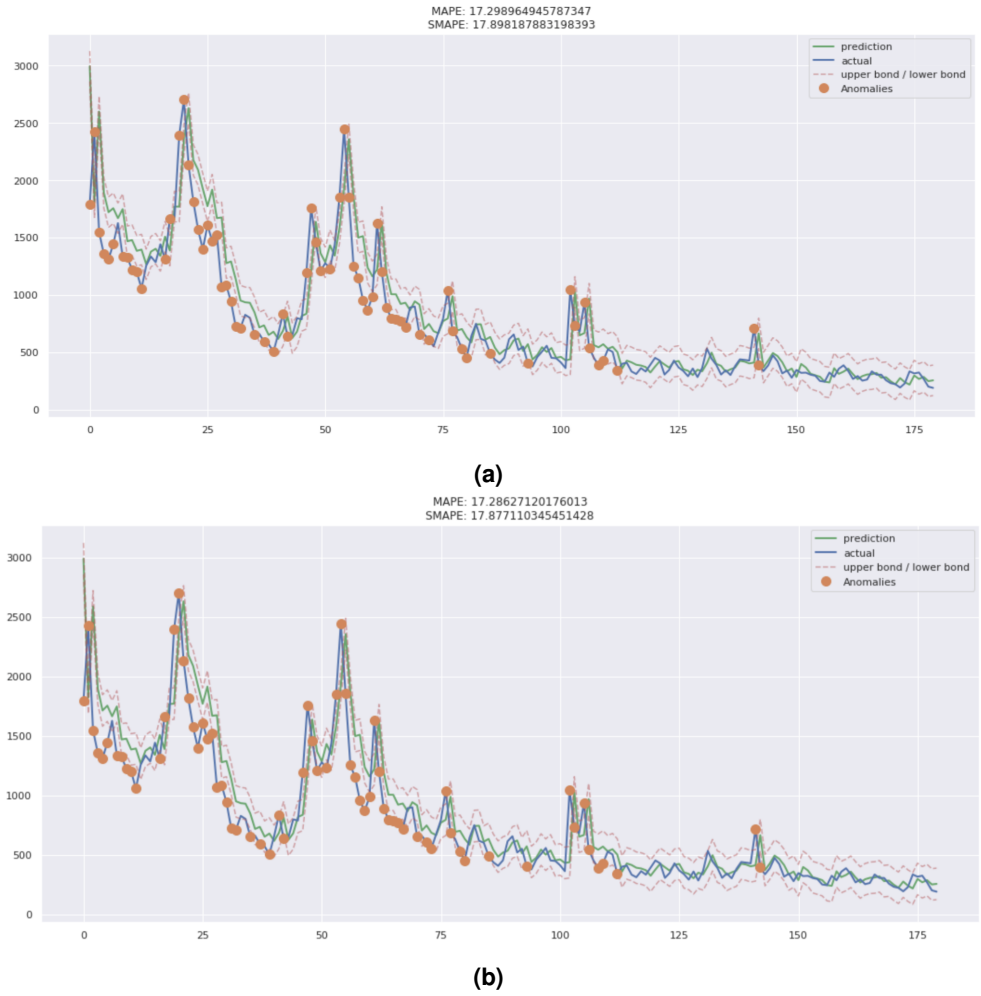


Figure 16. Predictions and ground truth values of the linear regression model trained with (top) Ridge regularization; (Bottom) Lasso Regularization

4.10 LSTM

4.11 Prophet

Enabling prophet to recognise long-term seasonal variations in the data is crucial for a successful forecasting of our time series data.

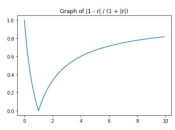


Figure 17. L2 regression

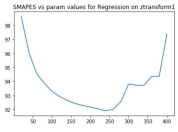


Figure 18. L2 regression

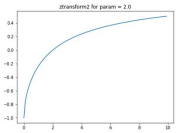


Figure 19. L2 regression

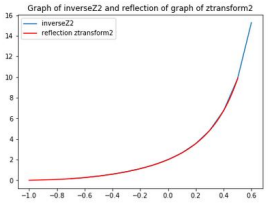


Figure 20. L2 regression

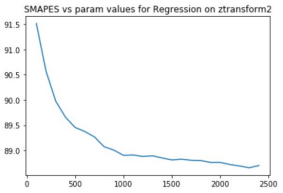


Figure 21. L2 regression

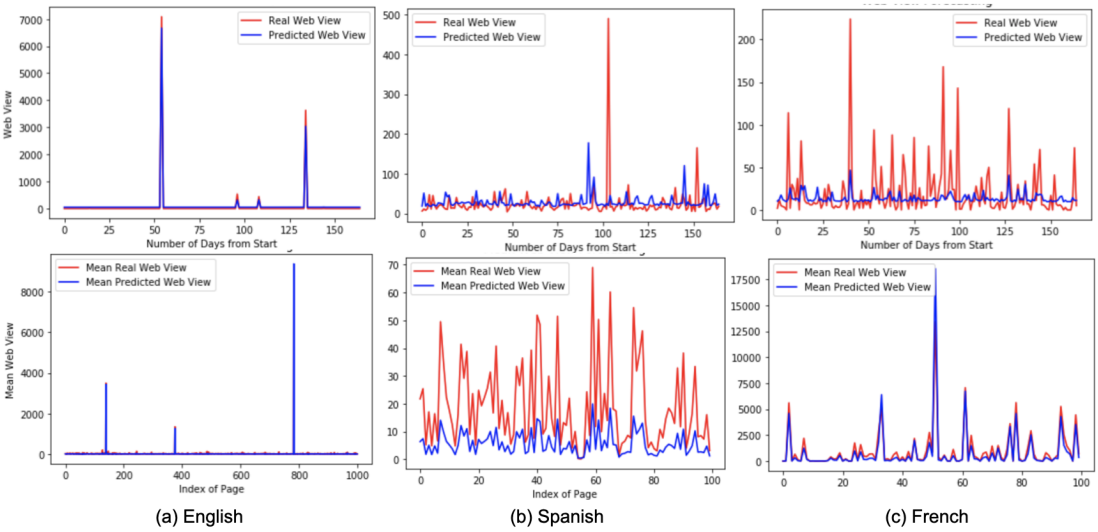
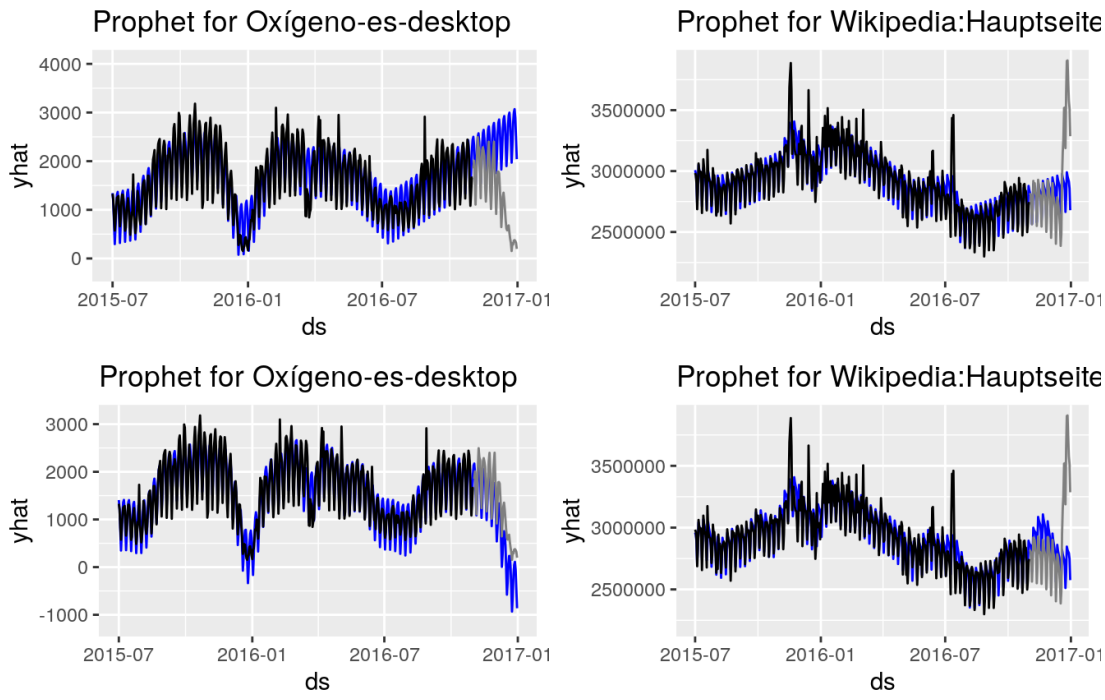


Figure 22. (Top) Web traffic forecasting versus number of days from start using LSTM (Bottom) Mean web view forecasting versus number of days from start using LSTM



The upper row of plot shows forecasts without a seasonal component vs the presence of this component in the lower row. We can clearly see that the seasonal forecasts predict the real time series evolution much better than the others. A seasonal component should be included in a successful prophet model for this project.

5 DISCUSSION

We observe that LSTM is the best forecasting model for predicting web traffic on different Wikipedia pages. The data accounts for regional and language differences. Using LSTM, we have forecasted the number of web views.

Model	SMAPE
ARIMA	0.65757
LSTM	**0.7235**
Simple Median Model	0.50096
Facebook prophet model	0.52810

However, we observe a spike in the real web views on certain days when predicted behavior is quite the opposite. Overall, the predicted result is quite accurate. We obtained a good accuracy when the LSTM model was trained on one page and tested on another, including the prediction of hikes in web views.
In the next experiment to improve SMAPE scores, we use properties of SMAPE transformation to guide us to some transformation $z = f(y)$ that will allow us to minimize SMAPE scores for L2 regression. As a result of these transformations, we observe an improvement in the SMAPE scores as given in the table below-
Some other models we have tested are Adaboost, Gradient Boost, and Random Forest. The results

Transformation	SMAPE scores
None	109.9994
z transform 1	91.49704
z transform 2	88.33296

Table 2. SMAPE score comparison

obtained have been presented below.

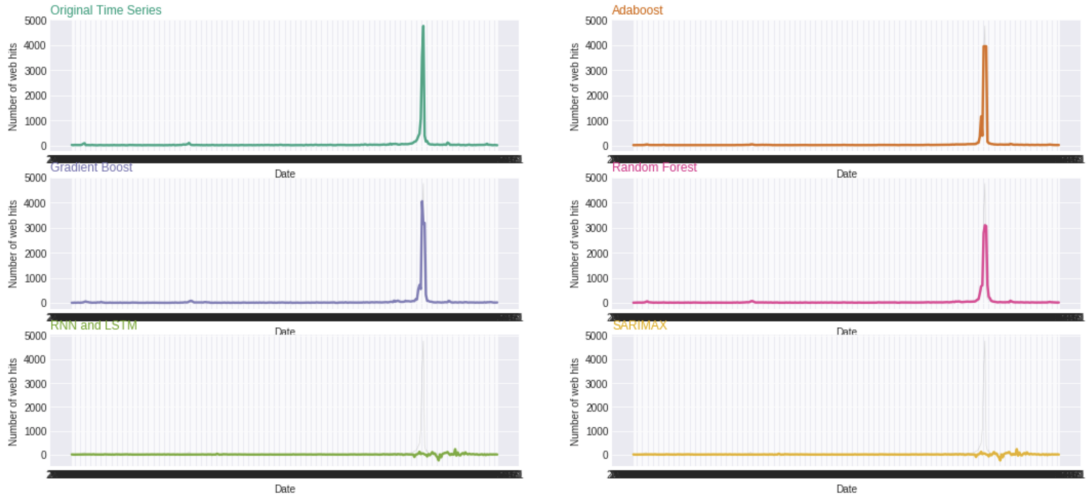
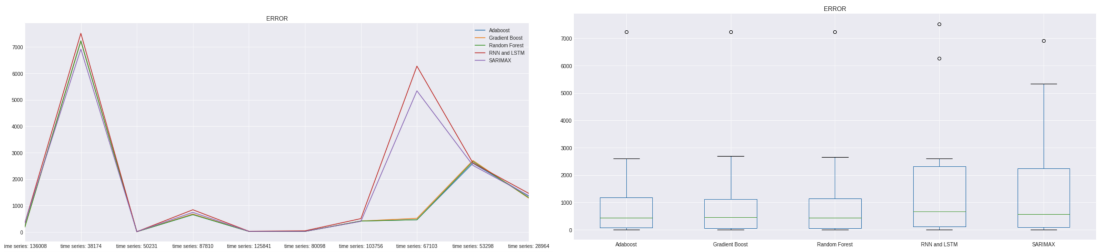


Figure 23. Prediction results of different models



(a) **(b)**

Figure 24. Error Comparison of different Models Based on RMSE value

	ME	RMSE	MAE	MPE	MAPE	MASE
Training set	-4.959912	79.11153	61.29358	-2.234391	9.799773	0.6007608
Test set	43.091885	197.43981	149.98938	-1.647010	22.237834	1.4701010

Table 3. ARIMA(2,1,2)

6 CONCLUSION

Time series forecasting is a growing area of research. The research done in the field yet is inadequate and various models are being developed to improve the accuracy for forecasting future values of multiple times series. In our term paper, we try to predict future web-traffic of Wikipedia pages. The purpose of the paper was to use past Values to predict future values. But time-series forecasting is important and has many important applications including resource optimization and congestion control.

In this study, we tested and analyzed various Machine learning and statistical techniques for the task of web traffic forecasting. We compare the outputs of these various models on different metrics and conclude that, in our study, using LSTM is recommended. Web traffic Time series prediction can be carried out using Long Short Term Memory Recurrent Neural Network and Auto-regressive integrated moving average more efficiently and accurately. LSTM makes the system more efficient and effectively captures seasonal patterns and long-term trends including information about holidays, day of week, language, region help our model to capture more correctly the highs and lows. Future works of this study include testing and analyzing the theoretical reasoning behind these results.

7 ACKNOWLEDGMENTS

Authors are thankful to Prof. S. Dharmaraja for their support and guidance and giving us the opportunity to attempt this term paper.

REFERENCES

Casado-Vara, R., Martin del Rey, A., Pérez-Palau, D., de-la Fuente-Valentín, L., and Corchado, J. M. (2021). Web traffic time series forecasting using lstm neural networks with distributed asynchronous training. *Mathematics*, 9(4):421.

Chen, M. (2011). Short-term forecasting model of web traffic based on genetic algorithm and neural network. In *2011 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC)*, pages 623–626. IEEE.

Petluri, N. and Al-Masri, E. (2018). Web traffic prediction of wikipedia pages. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 5427–5429. IEEE.

Wikipedia contributors (2021). Wikipedia — Wikipedia, the free encyclopedia. [Online; accessed 20-March-2021].

Yao, S., Hu, C., and Sun, M. (2006). Prediction of web traffic based on wavelet and neural network. In *2006 6th World Congress on Intelligent Control and Automation*, volume 1, pages 4026–4028. IEEE.

Zhou, K., Wang, W., Huang, L., and Liu, B. (2021). Comparative study on the time series forecasting of

web traffic based on statistical model and generative adversarial model. *Knowledge-Based Systems*, 213:106467.