Data Analytics: Project 1

Daniil Terekhin and Riccardo Giacometti

12 April 2023

1 Introduction

This data originates from blog posts where the raw HTML-documents of the blog posts were crawled and processed. The main task associated with the data is the prediction of the number of comments in the upcoming 24 hours according to total number of comments in different period, the length of posts, day of publication and other statictics of these features using Python and Data Analytics libraries.

2 Data Exploration

The descriptive statistics for the number of comments (target) are computed and shown in Table1. It can be seen that the range between the max and min is large and the median is zero. This can be further analysed by creating value counts for the number of comments where the top 5 is shown in Table 2. It can be seen that a large proportion of blog posts have zero comments which causes a large skewness in the data and reducing the number of zero comments should be considered. The large amount of blog posts with zero comments also explains why the median is zero since the median is sensitive against heavy skewness such as this. In addition, the percentage of comments that are zero is around 64%.

Table 1: Descriptive statistics of the target

mean	median	standard_dev	min	max
6.764719	0.0	37.706565	0.0	1424.0

Table 2: Frequency counts of each unique value from number of comments

Value	Frequency
0.0	33559
1.0	5775
2.0	2820
3.0	1654
4.0	1120

The Pearson correlation between the number of comments and all other features was also calculated and the top 10 correlations are summarised in Table 3. The correlations look promising however, after plotting the top correlation (column index 9) as shown in Figure 1, the high correlation value is misleading. This means that another approach needs to be taken.

Table 3: Top 10 correlations between number of comments and all other features

Feature	Correlation
9	0.506540
20	0.503375
5	0.497631
4	0.491707
10	0.490111
14	0.489674
19	0.486316
0	0.485464
number_comments_24_before_basetime	0.472061
15	0.471999

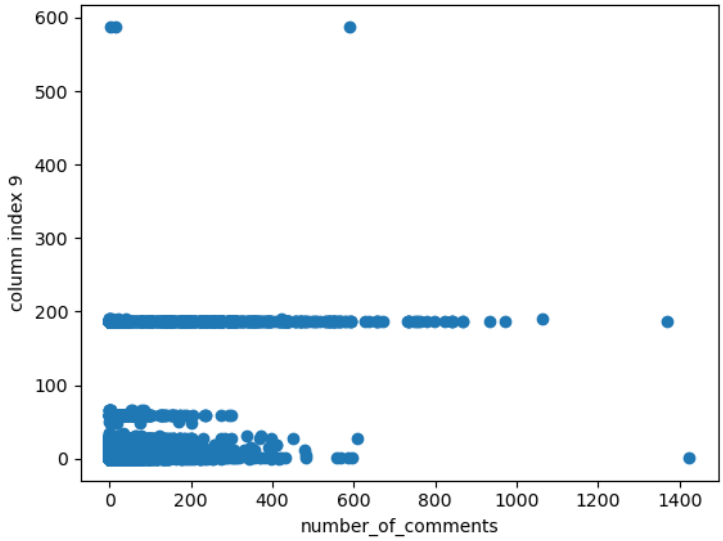

Figure 1: Example of plot from misleading correlation

Another interesting feature is the length of the post where the histogram of this feature is shown in Figure 2. It can be seen that most posts are short due to the large spike in the first bin from the left.

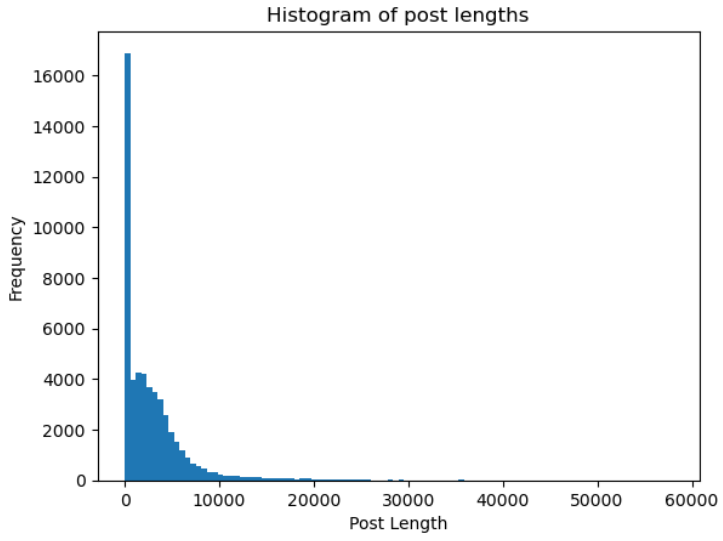

Figure 2: Histogram of post length

In addition to the post length, the day of publication is also analysed. The number of comments were grouped by the day of the week and samples with zero comments were excluded. Figure 3 shows a bar chart of the average number of comments aggregated by day. It can be seen that the day where one gets the highest average number of comments is Tuesday which is surprising since it was expected that people will read and interact with blogs in their free time (e.g Friday evening, Saturday, Sunday). In addition, the differences between different days is relatively small so it is not expected that the day of the week will be an important feature in the final model.

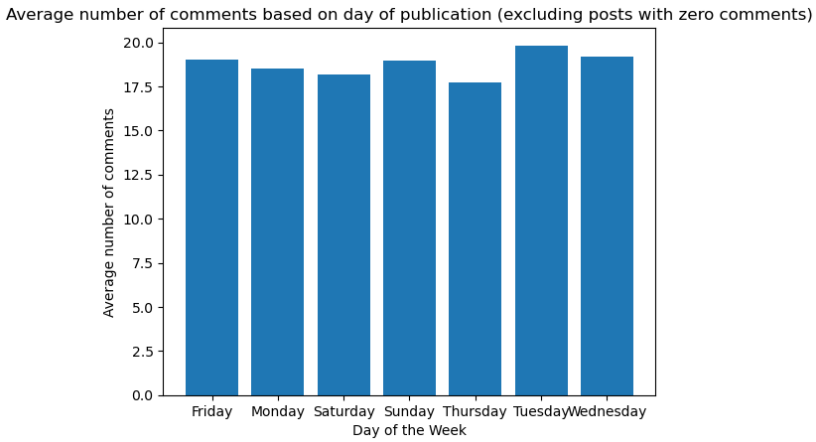

Figure 3: Average number of comments based on day of publication

In addition to the average number of comments by day, the maximum number of comments by day is also plotted. This will serve to show which days will yield a high number of comments. It can be seen from Figure 4 that the highest comments generated on a blog post happened on a Friday which is expected since users might be more likely to engage closer to the weekend. It is surprising however that the maximum number of comments for Saturday and Sunday are lower in comparison to their weekday counterparts.

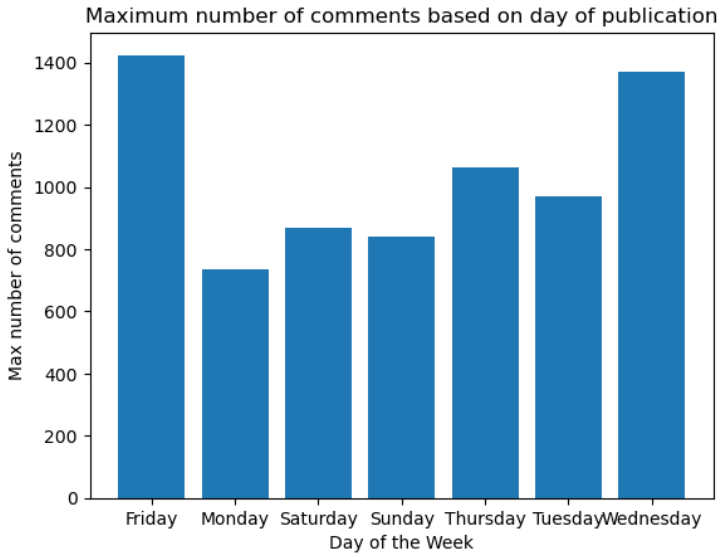

Figure 4: Maximum number of comments based on day of publication

Another feature that is analysed is the bag of words binary vectors with a vocabulary of the top 200 most common words. Table 3 shows the top 10 words that generate the highest average number of comments. It is not possible to check which column index represents which word therefore the index is given, in the case of the first index (84) this would correspond to the 22^{nd} most common word since the bag of words features start at column 62 (assuming index starts at 0). From this table, it is expected that the top 3 words should have some predictive power.

Table 4: Top 10 words with highest average number of comments

Column index of the word	Average number of comments
84	34.687500
91	31.230769
170	30.340909
153	23.350029
124	23.278182
136	20.917874
97	19.863636
217	19.275862
226	17.172205
194	16.678472

3 Data Pre-processing

As seen in the previous section, 64% of the number of comments column is zero so the subset where the number of comments is zero is reduced to 6000 through random sampling. In addition, since there are blog posts with a large number of comments, the maximum number of comments is restricted to 35. This helps to have a better data distribution. Figure 5 and 6 compare the distribution of the number of comments before and after processing the data, respectively.

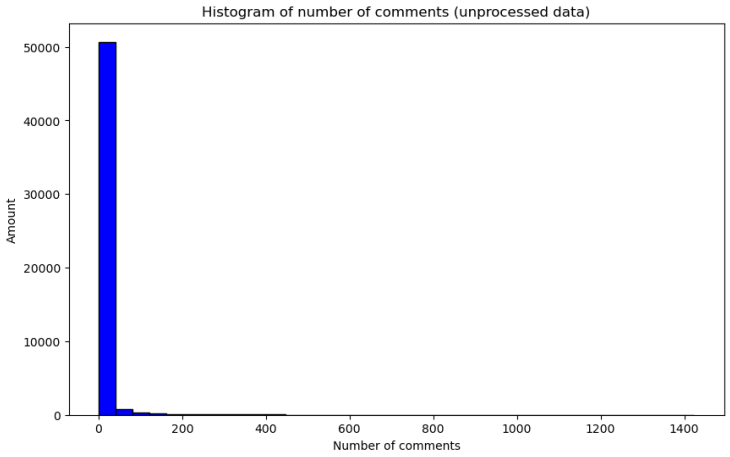

Figure 5: Distribution of number of comments before processing

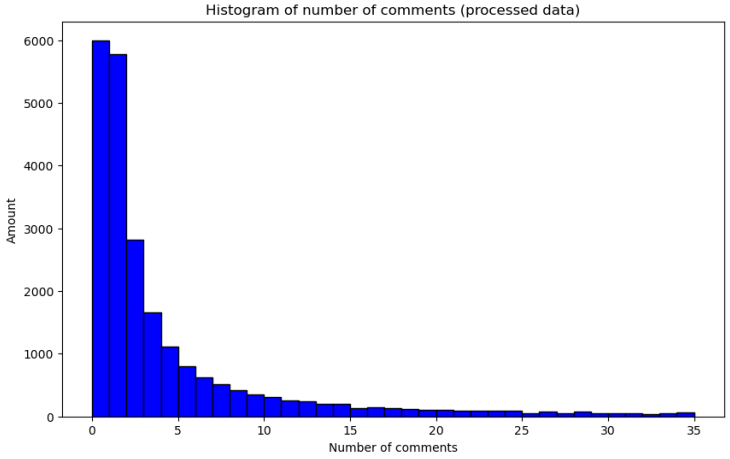

Figure 6: Distribution of number of comments after processing

4 Model

For prediction of a target value we tried to use several models and evaluate their performance with Mean Absolute Error (MAE) and Mean Squared Error (MSE). We used Linear Regression, Random Forest Regressor, Gradient Boosting Regressor and XGBoost Regressor.

```
Linear Regression               MSE: 27.94637935531971   MAE: 3.362424965464046
Random Forest Regressor         MSE: 25.580158520692056   MAE: 3.052972264330567
Gradient Boosting Regressor     MSE: 13.87266039569145   MAE: 2.2626557522078588
XGBoost                         MSE: 13.770687203342323   MAE: 2.2425372013366003
```

Figure 7: Perfomance of 4 models on train data

We can see that XGBoost Regressor had the lowest MSE and MAE, so we evaluated this model on test data. The result was 2.67. From feature importance we got 10 columns with the biggest value on our target column.

	features	importances
0	number_comments_24_before_basetime	0.032049
1	9	0.017350
2	difference_comments_48_24_and_24_before_basetime	0.013521
3	20	0.013492
4	time_between_publication_and_basetime	0.010913
5	14	0.010212
6	19	0.009351
7	saturday_publication	0.009323
8	10	0.009143
9	246	0.009054

Figure 8: Feature importance of XGBoost Regressor model

We can notice that feature number of comments 24 hours before basetime has the most value on number of comments in the upcoming 24 hours (our target). Saturday as a day of post publication also quite impactfull for prediction.

5 Discussion and Conclusion

Even though the XGBoost model outperformed the other models tested, it still has a relatively high error. Given that the average number of comments is 6.76

and the MAE on the train data set is 2.19 this means that it the percentage error range relative to the average number of comments is 64.7% which is very high and on the test data set it performs worse with an MAE of 7.54. It is suspected that the large error originates from a few factors.

The first one is that due to the 200 bag of words vector and the day of the week vector, the sparsity of each sample is significant which increases the likelihood that the model will fit the noise in these features. In addition, the data contained 280 features which might be too much also considering that a significant proportion of these features will have little to no predictive capabilities. Using the top 10 importance features from XGBoost was briefly tested in order to use the features with the most predictive power and improve generalisation however, this yielded similar results to the XGBoost model that employed all the features. It could also be possible that the data along with its features could be deemed as a non-ergodic system and therefore is hard to predict unless other features are included. An example of this could be perhaps the general subject of the blog (i.e blogs about food might have more user engagement compared to others) where features such as these might be able to yield more predictive capabilities.

In conclusion, a model to predict the number of comments over the next 24 hours in a blog post was developed after testing other various models. The overall predictive power of the model was weak due to the suspected pitfalls mentioned before.