CMPT353 Project Report

Leo Li, Joao V. Ishida Correa
301372909, 301449583

The Problem

The problem we are addressing is what makes a Reddit submission good or bad. To solve this problem, first, we have to define what is a good submission. Understanding what makes a submission successful on Reddit involves analyzing various factors, including user engagement, timing, the day of the week, submission score, readability, and sentiment. A Reddit score is the number of upvotes minus the number of downvotes. A fixed score threshold would be unfair for small Subreddits with less popularity than the large ones. Therefore, we consider the submission with the top 50% score within its Subreddit to be a good submission.

The Data

Gathering and cleaning the data

The data is gathered from `reddit-3` on the cluster. We only care about submissions, not comments, so we used `extract_subsets.py` to extract all the submissions from the five medium-volume subreddits for 2016. We moved the data from HDFS to our computers so we can work on it locally. Spark has an overhead problem on a relatively small dataset. We used `transfer_to_csv.py` to transfer the data from Spark Dataframes to CSV files. We used Python, Pandas Dataframes, and CSV for the following research.

To clean up the data, we converted the created_utc to date time for easier reading and replaced the 'media' link with 'is_media' since knowing the hyperlink of media is not useful, but knowing whether a submission contains media or not is useful. We removed the 'ups' column because it has the extract same information as the 'score'. We dropped columns that only contain 0s or NaN, for example, 'archived', 'downs', and 'hide_score'. Then we removed columns with insignificant values, for example, only 2 out of 15005 entries have 'gilded' more than 0, and 9 out of 15005 entries have 'over_18' equal true. The sample is too small to make an impact in our research, so we dropped those columns. Lastly, we dropped the irrelevant columns such as 'author_flair_css_class' and 'permalink'.

Modification

To add a column 'good' that has a boolean type that indicates the submission is good, we calculated the 50th percentile score for each subreddit. The result further confirmed that a single standard score to determine submission across different Subreddits is unfair.

subreddit	score
Cameras	1.0
Genealogy	4.0
optometry	1.0
scala	8.0
xkcd	18.0

Furthermore, we joined the two dataframes and compared them to get ‘good’.

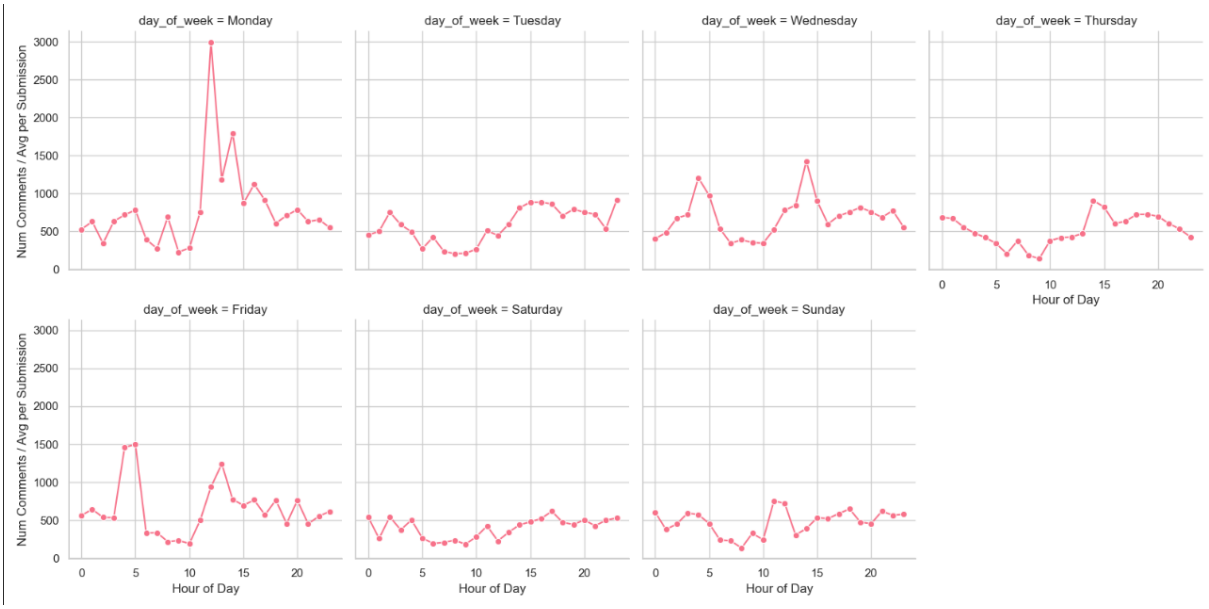
After the cleaning and modification process, the data has labels ['distinguished', 'is_self', 'month', 'num_comments', 'score', 'selftext', 'subreddit', 'title', 'year', 'timestamp', 'is_media', 'good'].

Comparing the Number of Comments with the Date of Submissions

First Analysis

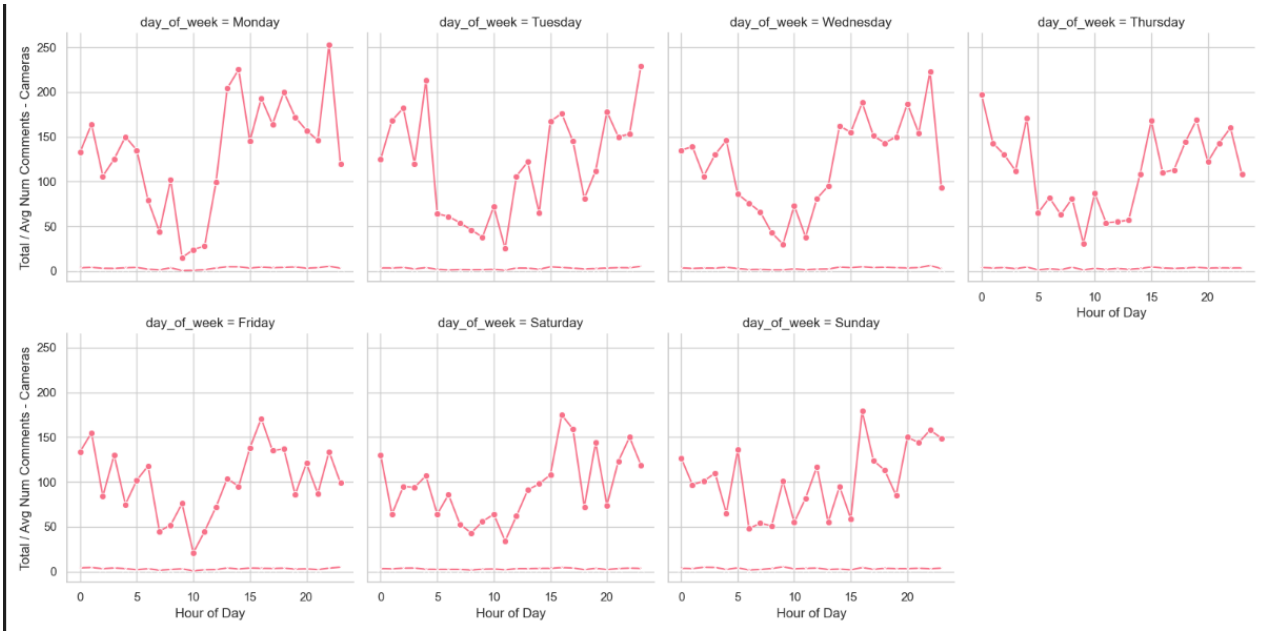
We used Python with Pandas, Matplotlib, and Seaborn libraries to explore the relationship between the number of comments and the time of submission on Reddit across five diverse subreddits: cameras, genealogy, optometry, scala, and xkcd.

Our analysis involved examining the average submissions per hour per day of the week for different subreddits, aiming to identify patterns that distinguish successful submissions. The code allowed us to examine this data, providing insights into what might constitute a well-engaged post.



In the small multiple-time series plot above, we could identify that different hours of the week had distinct numbers of comments for the posts made during those specific times. This visualization allowed us to discern the varying engagement levels across different hours of the week. However, while it helped in identifying time-based trends in comment counts, it wasn't sufficient to conclusively determine whether a certain posting time could directly categorize a submission as 'good' or 'bad' based on comment engagement alone.

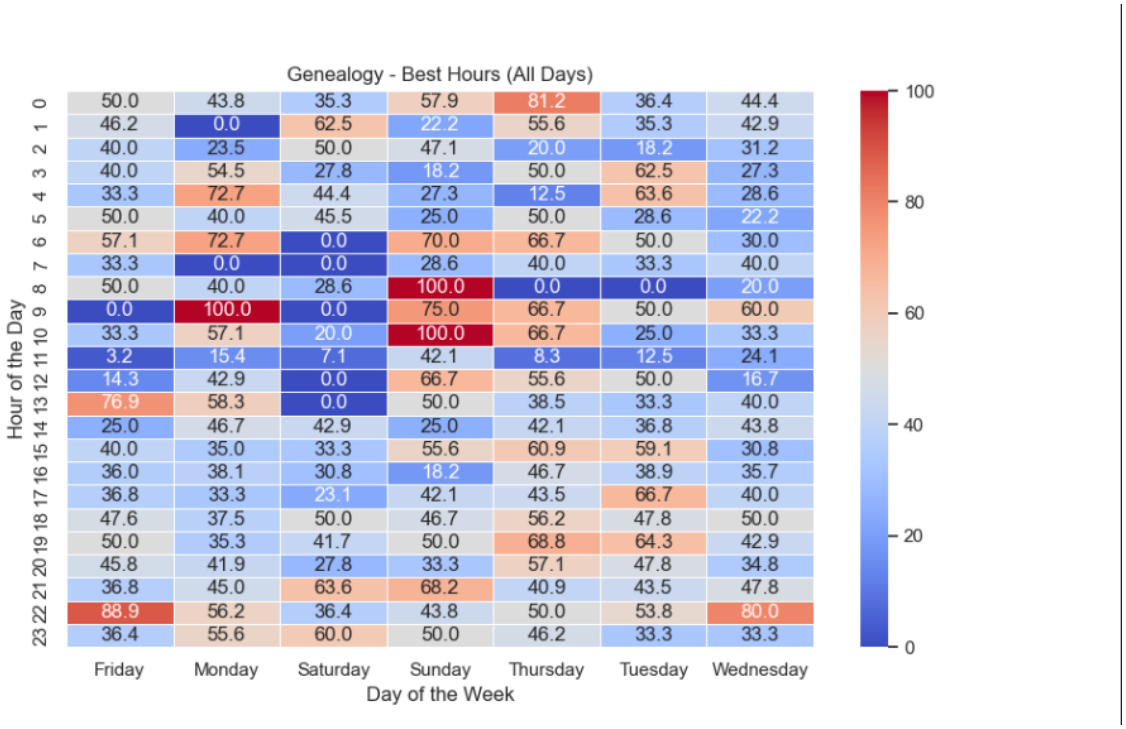
This observation was consistent even when attempting to isolate specific subreddits for more targeted analysis, such as 'Cameras' below. Despite segregating data by subreddit, the analysis did not yield a definitive conclusion on whether a certain posting time could consistently indicate a submission's quality as 'good' or 'bad'. Although the data revealed fluctuations in comment engagement across different hours and days, it didn't directly align those fluctuations with a binary categorization of submission quality.



Getting Results

Later, an incremented analysis based on submission scores and hours of creation was conducted. The code organized submissions, analyzing their labels of ‘good’ or ‘bad’ by comparing their comment engagement against the average comments per hour to verify if the label has anything to do with the time or an odd outlier. Heatmaps were also created to visualize optimal and less favorable posting times for each day, displaying the percentage of submissions meeting criteria for 'good' or 'bad' engagement based on comments per hour.

Analysis of these results highlighted peak hours during each day, indicating when a higher proportion of submissions received comments above the hourly average, defining them as ‘good’. Conversely, off-peak hours represented lower comment activity, identifying these time slots as ‘bad’ for submissions.



In summary, this analysis revealed insights into how posting times influence user engagement for Reddit submissions. This analysis is among several methods used to evaluate what makes a Reddit submission successful or less successful. This combined approach aims to offer Reddit users a comprehensive understanding of what defines successful submissions and guides how to time their posts for increased visibility and interaction.

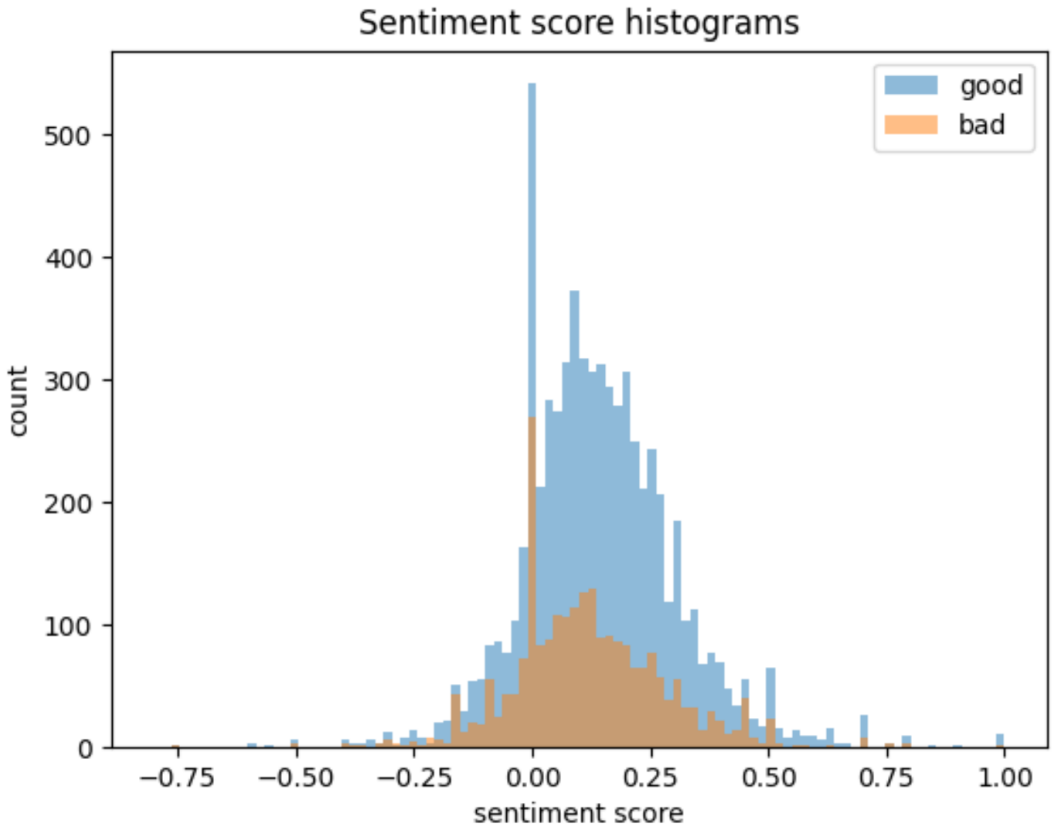
Sentiment Analysis

People may wonder that does the sentiment of the text in the submission affects the submission be good or bad. We are going to find out in this part.

Many submissions do not have text within it. Some only contain a media hyperlink, and some are [delete] or [removed]. We excluded them for sentiment analysis. We used the library TextBlob to find the sentiment score. The TextBlob takes a paragraph of text as input and output ‘polarity’ and ‘subjectivity’. We believe knowing there are personal opinions or fractal information in a

submission is not as important as knowing the polarity of the sentiment in this study. Therefore we only take the 'polarity' as the measurement of sentiment. The 'polarity' has a range of [-1.0, 1.0]. -1.0 means the text has an extremely negative sentiment. And positive value means positive sentiment.

We calculated the sentiment score for every entry. We split the data using 'good', so we could compare the good submissions' sentiment score with the bad submissions' sentiment score.



We are going to conduct a T-test on the data. The null hypothesis is the mean of the sentiment score for good submissions and bad submissions are the same. To prepare the T-test, we check the normality.

```
Normality p-value for good submissions :  2.2511514704695988e-139
Normality p-value for bad submissions:  1.756037676034935e-40
```

They did not pass the normality test because the kurtosis of the data was too high.

```
Kurtosis for good submissions :  2.815678168663144
Kurtosis for bad submissions :  2.7634924644053456
```

However, since we have a lot of data points (good > 6000, bad > 2000), based on the central limit theorem, we argue it is probably okay to continue to T-test.

```
Equal variance p-value for good vs bad submissions :  0.24753521712111243
```

We conducted the equal variance test and the data passed it.
As we satisfied the requirement for a T-test, normally distributed and has equal variance, here is the result for the T-test.

```
T-test p-value for good vs bad submissions:  1.848842304211237e-09
```

The p-value is smaller than 0.05. We reject the null hypothesis. We concluded that the good submissions have a different sentiment score mean than the bad submissions.

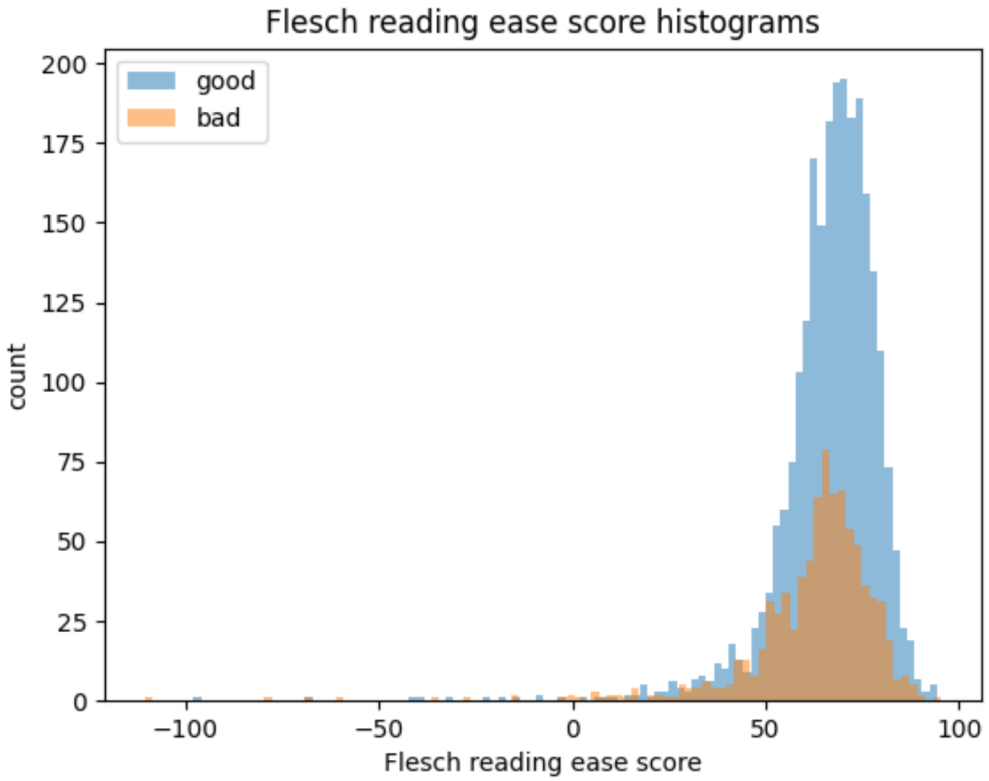
```
Good submission mean:  0.1419238662745762 Bad submission mean:  0.11763245455116916
```

The good submissions have a higher mean of sentiment scores than bad submissions.

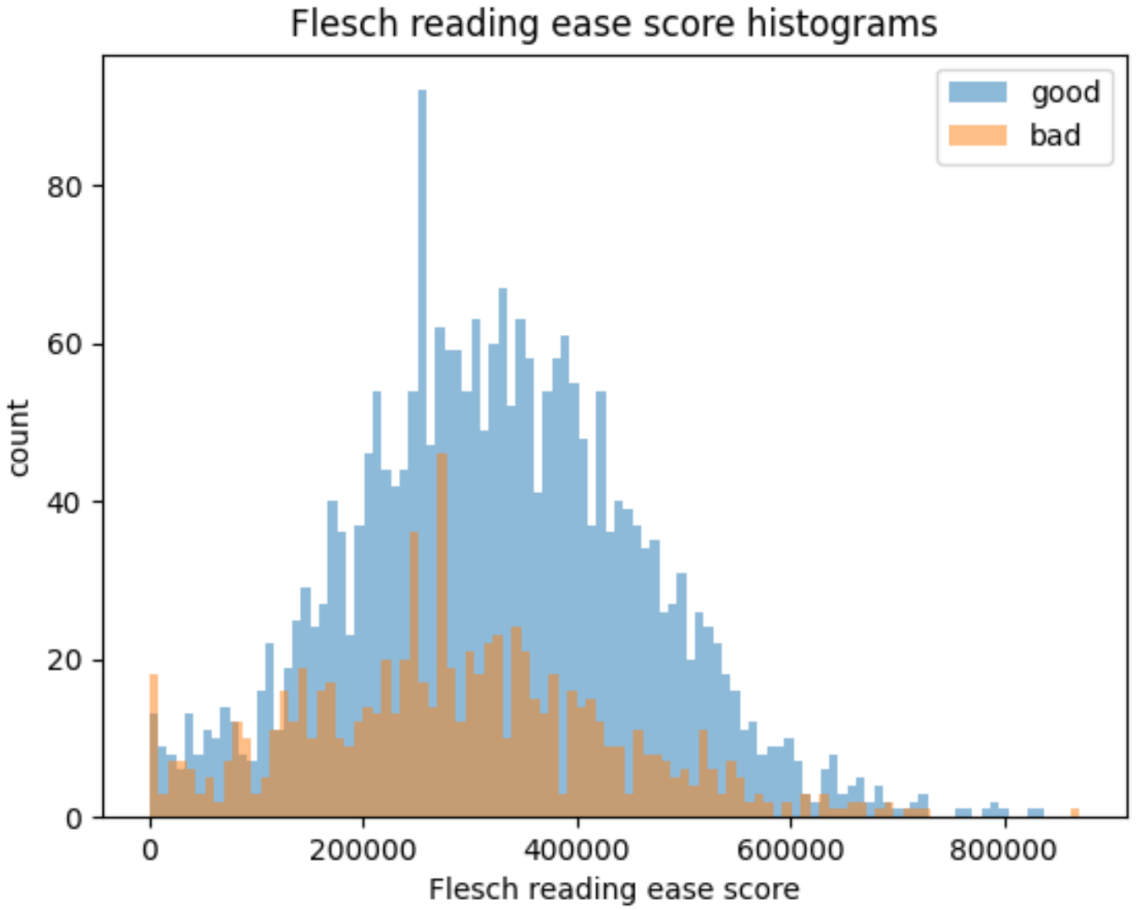
Readability Analysis

The readability of the text in a submission could be a factor in making that submission good. The library we used to determine readability is [py-readability-metrics]. We input a paragraph of text with a minimum of 100 words, and the Readability() outputs the Flesch Reading Ease score. The Flesch Reading Ease score has a maximum of 100. If the score is 100, it indicates the text is very easy to read and understand. A score of 0 means the text has a very bad reliability. However, there is no minimum for the Flesch Reading Ease score. It can be negative.

The Readability() requires input to be text with a minimum of 100 words. We excluded entries with no text or [delete] or [removed] only. We further filtered out entries with text less than 100 words. We calculated the readability score for each row and split the data based on 'good' like the previous analysis.



The above shows a histogram of the readability score for each group. We decided to remove some outliers. We decided every negative score are outlier. The histogram looks left-skewed. Therefore, a data transformation was applied. We used x**3 to reduce the left-skewed problem.



After removing outliers and data transformation, the data looks ready for the T-test. The null hypothesis is the mean of the readability score for good submissions and bad submissions are the same.

Checking normality:

```
Normality p-value for good submissions :  0.00071792911717201 53
Normality p-value for bad submissions:  0.0007339518463553961
```

The data did not pass the normality test. But again, since we have a lot of data points, based on the central limit theorem, we argue it is probably okay to continue to T-test.

Checking equal variance:

```
Equal variance p-value for good vs bad submissions :  0.25872112598294733
```

The data passed the equal variance test.

With normality and equal variance pass, here is the T-test p-value:

```
T-test p-value for good vs bad submissions:  1.819755680659184e-11
```

The p-value is smaller than 0.05. We reject the null hypothesis. We concluded that the good submissions have a different readability score mean than the bad submissions.

```
Good submission mean:  66.61089300531268 Bad submission mean:  62.147923086317036
```

The good submissions have a higher mean of readability scores than bad submissions.

Conclusion

In summary, both sentiment and readability were identified as influential factors in determining the success of Reddit submissions. "Good" submissions were associated with higher sentiment and readability scores compared to "bad" submissions. However, at the time of submission, multiple factors could affect this. It was found that for different subreddits, different hours of the day the week had different responses to their ‘good’ or ‘bad’ associations. Although, it is also safe to say that within each subreddit, there are hours and days of the week that can affect if the submission is concluded to be ‘good’ or ‘bad’. To answer the problem statement, we can safely say that high sentiment and readability scores contribute important factors to make a Reddit submission good and that hours of different days of the week contribute to the submissions within their subreddit environment.

Limitations

We did plan to try different machine-learning models for predicting whether a submission is good/bad, and comparing the accuracy and analyzing them. We had the framework finished in the `model_accuracy.py` file, but we did not have enough time to complete and test it, so we could not present it in the report.

Leo’s project experience summaries

- Utilized PySpark on the cluster to extract and process Reddit submissions data by using techniques with Python, Pandas, and CSV to convert PySpark dataframes to Pandas dataframes and cleaned the data as a result of a cleaned data with relevant and significant values that can be used for the analysis phase.

- Leveraged external Python libraries such as TextBlob to do data analysis on the impact of sentiment and readability on what makes Reddit submissions good/bad. Conducted statistical tests such as the Normality test and T-test to reach the statistical conclusion. Utilized graphing library matplotlib to produce a visualization of data for providing a better understanding of the data.

- Filtered out submissions with insufficient text and applied data transformation with Pandas to address left-skewness and conducted a T-test on readability scores, uncovering a statistically significant difference between good and bad submissions.

Joao’s project experience summaries

- Used Python tools like Pandas, Matplotlib, and Seaborn to understand Reddit submission data. Identified 'good' and 'bad' submissions based on engagement metrics. Created visuals and more than 1000 lines of numerical analysis, showcasing best posting times for enhanced submissions.

- Leveraged Seaborn's capabilities to create over 10 informative plots, revealing engagement trends across subreddits and posting times. Showed optimal and less favorable posting periods, guiding users to improve their Reddit submissions' engagement.

- Developed detailed reports outlining submission analysis, findings, and recommendations. These resources explained methodologies, key results, and their implications, aiding informed decision-making and optimizing posting strategies on Reddit.