Distributed Computing Final Presentation

William Guenneugues
Anthony Wang
Lawrence Lin

Project Overview:

- In this project we set out to determine if social media sentiment of a controversial content creator/artist can influence their viewership/sales.
- For our case study, we chose to focus our attention on Joe Rogan, a renown sportscaster and number one podcaster in the world.

The data:

For this project we gathered data from three separate sources:

- Twitter: Using the snscrape library, we scraped 80 Mb worth of data, amounting to over 400,000 tweets posted in the last 500 days.
- Reddit: Using the reddit API, we scraped 15 Mb worth of data, gathering posts made in the last 500 days.
- Youtube: We scraped data from the website socialblade to gather the weekly viewership of Joe Rogan’s channel between November 2020 and now.
- After gathering all the data, we uploaded it to S3.

Preprocessing Algorithms

Using Spark, we did following preprocessing operations:

- Unioned twitter and reddit spark dataframes, joined on date.
- Added Youtube viewership data as a column, joined on date
- Feature Engineering
    - Text length (Count of characters): ran in 0.45 seconds
    - Emotion label (using Roberta English model): ran in 2.78 hours
    - Sentiment 1 score (using VaderSentiment model): ran in 17.47 minutes

Time Efficiency

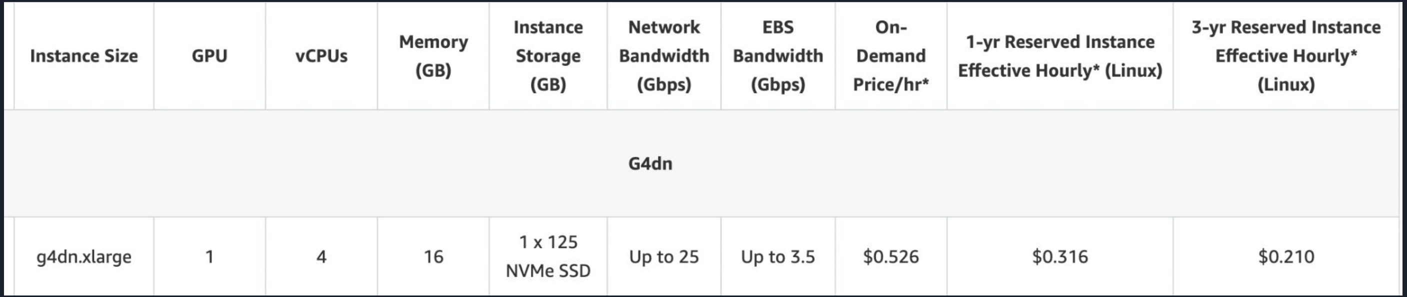
Time for all pre-processing algorithms: 4.02 hours

Time to train all models: 3 minutes

- Optimizations:
    - Cached dataframe
    - Running individual group GPU
    - Added index on date on MongoDB

Machine Specs

We used a shared GPU with the following specs:



Instance Size	GPU	vCPUs	Memory (GB)	Instance Storage (GB)	Network Bandwidth (Gbps)	EBS Bandwidth (Gbps)	On-Demand Price/hr*	1-yr Reserved Instance Effective Hourly* (Linux)	3-yr Reserved Instance Effective Hourly* (Linux)
G4dn									
g4dn.xlarge	1	4	16	1 x 125 NVMe SSD	Up to 25	Up to 3.5	$0.526	$0.316	$0.210

Final Preprocessed Data Schema:

After running all the preprocessing operations, we uploaded data to MongoDB with the following columns:

- Week (Datetime: start day of the week)
- Text (String: text from social media post)
- Likes (Integer: number of likes, only available for twitter data)
- Views_gained (Long: number of views gained/lossed compared to previous week)
- Text_length (Integer: character length of social media post)
- Hf_emot_label (String: Emotion Label of text: Joy, Sadness, Fear, Anger, Love, Surprise)
- Vader_sentiment_score (Float: Sentiment score from -1 to 1, with 1 being most positive)

Machine Learning Model 1: Linear Regression

Linear model coefficients:

Likes	Text Length	Vader Sentiment	Hugging Face Emotion Dummy variable
28.6	-224.21	19472.38	-294766

- Linear model MSE: 8.47E5
- Most important feature: emotion and sentiment scores

Machine Learning Model 2: Random Forest

Linear model coefficients:

Likes	Text Length	Vader Sentiment	Hugging Face Emotion Dummy variable
0.5052	0.1427	0.0687	0.1122

- Random Forest Regressor MSE: 8.47E5
- Most important feature: Likes and text Length

Machine Learning Model 3: Gradient Boosting

Gradient Boosting Feature Importance:

Likes	Text Length	Vader Sentiment	Hugging Face Emotion Dummy variable
0.4017	0.2927	0.0288	0.2768

- Gradient Boosting Regressor model MSE: 8.47E5
- Most important feature: Likes, text Length and Emotion

Machine Learning Outcomes:

- All models gave us the same normalized MSE value.
- The different models gave us different feature importance values.
- Linear regression works well as a quick and easy baseline model w/o the need for hyperparameter tuning

Lessons Learnt:

- Sharing a cluster with many people will slow down the operating speed.
- Run time was improved when we cached datasets.
- Optimizing code matters when it comes to big data; can’t afford to have slow or redundant code.
- Feature Importance is hard to determine and can vary a lot between different models.
- Some functions like the Hugging Face model took much longer to run than others.

Conclusion:

- Our best model was: Gradient Boosted Regression Trees, but the losses were virtually identical.
- Social media sentiment analysis: We found that some features had significant impacts on their respective models, but which features were important varied from model to model.
- Ultimately, because we determined some features to be significant, we can assert that social media sentiment is positively associated with Joe Rogan's viewership.