MINECRAFT: DIGGING FOR INSIGHTS

GOALS AND OBJECTIVES

- Expand and operationalize Twitter data sets
- Expand and explore new platforms for data collection
- Develop pipeline for automated topic modeling and sentiment analysis of Twitter data
- Collect and analyze YouTube comments on Minecraft updates
- Investigate additional social media platforms for data collection
- Establish automated weekly analysis of topics and sentiment for efficient processing

TWITTER

- Existing listener from the previous team built with python library Tweepy
- Listener collected Minecraft related Tweets everyday
- Previously, listener lacked a pipeline to perform topic modeling and sentiment analysis on

YOUTUBE

- Existing scraper from the previous team
- Old scraper relied on YouTube API
- Old scraper limited by YouTube API quota
- Old scraper limited to only a select few channels and creators of interest
- New scraper is more "generalized" and collects data about what is being most talked about

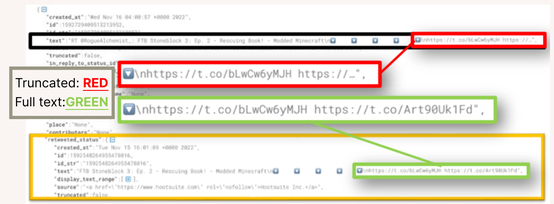
DATA ENGINEERING

DATA ENGINEERING - TWITTER

For Twitter, the data engineering work mainly consisted of working on the pipeline and automating tasks for weekly analysis.

The data engineering team needed to convert and save the data from the listener, coming in as a JSON format, into something more accessible for the data science team. This was done by writing a script to automatically parse the JSON and write it to Databrick's datastore where it could be used in the pipeline.

The data engineering team also set out to fix a bug with the Twitter listener where some tweets were being truncated. It was discovered that the listener was reading the Tweet status object returned by Tweepy incorrectly. Below is an example of a Tweet status object. In black is where the tweet text was previously extracted. In yellow is where the full text is now being correctly extracted.



DATA ENGINEERING - YOUTUBE

For YouTube, we wanted to develop a new scraper that did not rely on the YouTube API. We found the YouTube API limiting due to the daily quota (limits how much data we can collect) and lack of options to search for and find relevant videos. We decided to build our own scraper using Selenium that could freely scrape through videos without worries about limits.

In the end, we developed two scraper versions. The first version is a general scraper that will use search terms to find related videos. The second version is a creator-specific scraper that will scrape videos from a single channel of interest.

DATA SCIENCE

DATA SCIENCE

During the first semester, we chose to focus on Twitter because there was already an existing Twitter scraper from last year. This means that we could build on previous work and expand on the previous pipeline.

During the second semester, we chose to focus on YouTube because it is one of the most popular platforms for Minecraft. The previous scraper collected data from specific channels, and we wanted create something more general to Minecraft as a whole for the platform.

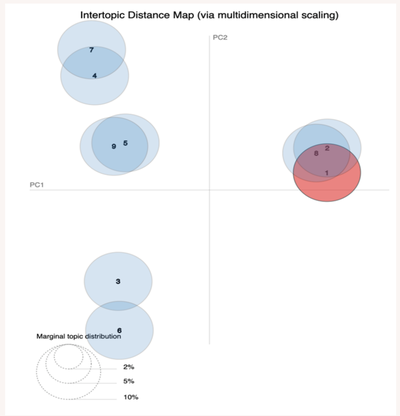
TOPIC MODELING - LATENT DIRICHLET ALLOCATION (LDA)

For topic modeling, we used Latent Dirichlet Allocation (LDA). LDA is an unsupervised clustering technique that is commonly used for text analysis. It's a type of topic modeling in which words are represented as topics, and documents are represented as a collection of these word topics.

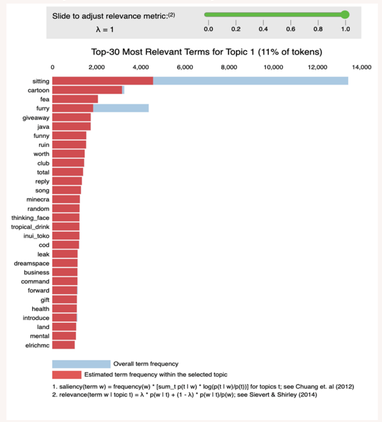
Why LDA?

- Can find latent topic inside documents
- Supervised learning requires a true label, which may not be available
- LDA is easy to train
- LDA give interpretable topics

TWITTER TOPIC MODELING RESULTS - OCT 19TH, 2022 – OCT 26TH, 2022



Each circle represents a topic that contains related words. The distance between circles is how closely related each topic is. The size of each circle is proportional to the number of words in each topic.



The bars represent the frequency of terms in a topic. This is showing terms for topic 1.

DATA PREPROCESSING

Data preprocessing work was done to clean up and make the data easier to interpret and work with the ML models.

The first preprocessing pass was done to translate emojis to text that could be using in NLP.

The second preprocessing pass was done to lemmatize and remove stop words from the data. The purpose was the further optimize the data for interpretation by removing low level information and normalizing text.

Lastly, work was done to translate text from other languages into English so that data from other countries could be used and interpreted.

😂 -> face_with_tears_of_joy
😊 -> smiling_face_with_smiling_eyes
😡 -> pouting_face

Example of emoji translation

Leaves -> Leaf

Ponies -> Pony

Lemmatization to reduce words to their base forms

"An apple a day keeps the doctor away:" -> "apple day keeps doctor away"

"This is the best" -> "best"

Removing stop words that do not add meaning

MINECRAFT YOUTUBE CHANNEL TOPIC MODELING RESULTS - MAR 21ST, 2023

With the data from the YouTube scraper, we used the same LDA model that we used for the Twitter data. The word clouds below are of the most brought up topics from the comments of the 28 most recent YouTube videos on the official Minecraft channel using our LDA model. While we started by using comments from general Minecraft videos, we found that too many creator-specific topics showed up. Instead, we decided to pivot to using videos from the official Minecraft channel as we felt that the videos stayed more on topic and we would get less creator-specific comments. However, we are still able to performing topic modeling using comments from general Minecraft videos.

SENTIMENT ANALYSIS

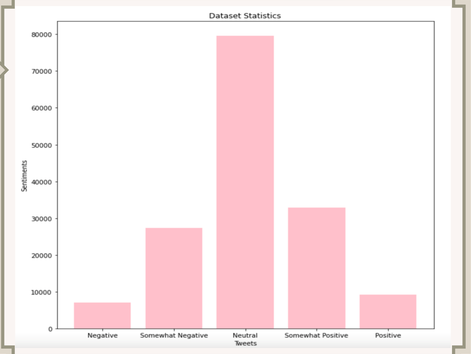
For sentiment analysis, we chose to use BERT (Bidirectional Encoder Representations from Transformers). We chose BERT because, as opposed to other RNN architectures, BERT models demonstrate state-of-the-art accuracy across various tasks.

```
input_text = 'Get ready to break those blocks! Spleef in just under an hour on #spleef #speedbuildisnextweek #nerdnu #minecraft #mcservers #mcpc'
processed_data = prepare_data(input_text, tokenizer)
result = make_prediction(sentiment_model, processed_data=processed_data)
print(f"Predicted Sentiment: {result}")
```

Accuracy attained: 89%

Sample tweet: "Get ready to break those blocks! Spleef in just under an hour on #spleef #speedbuildisnextweek #nerdnu #minecraft #mcservers #mcpc"

Result: Neutral sentiment



A SPECIAL THANKS TO... MATTHEW SKIRVIN AND THE DATA MINE STAFF

TAs: Veronica Fulbright **Students:** Aaron Ni, Arnav Arora, Yifei Jin, Arya Medapati, Eric De Leon, Sathvika Kotha, Ishan Rathod, Vaishnavi Rudraraju, Venkata Divya Sree Pulipati, Mahima Natarajan, Sweta Kesur, Mihir Mutyampeta, Rohit Kannan, Siddharth Singh, Hridhay Monangi