MINECRAFT:

A Natural Language Processing Project using the Minecraft Github Repositories

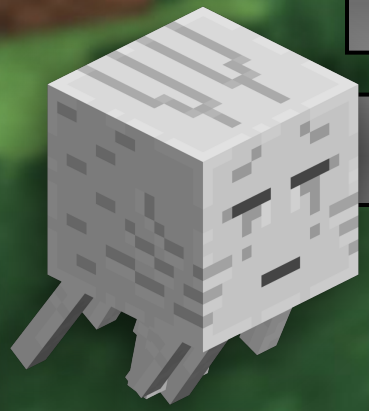
January 17, 2023

Chris Rosenberger
Cristina Lucin
Michael Mesa
Rae Downen

Start the Presentation

Options

Quit



Loading……

WELCOME

This project focuses on building a prediction model for accurately predicting the coding language of a project using examination of GitHub Minecraft Repo README files. Our goal is to develop several predictive models utilizing Python and Python libraries to select the most effective model for production.

Data Science Pipeline

Acquisition | Preparation | Exploration | Modeling

- 1,000 Repo URLs tagged "Minecraft" were acquired from GitHub utilizing a .py script "acquire_minecraft_urls.py"
- These Repos were identified and scraped through the search feature in GitHub
- Repo Readme Text and Repo Language was scraped utilizing BeautifulSoup
- Readme Text and Repo Language was collected into a dictionary using a function called "process_repo.py" and "scrape_github_data"
- This dictionary was turned into a dataframe and CSV file
- The CSV file contained 1,000 rows and 3 features before cleaning (884 after cleaning)
- Each row represents a unique Repo located on Github
- Each column represents a feature of the Repo, such as URL, Readme text, and Programming Language

Data Science Pipeline

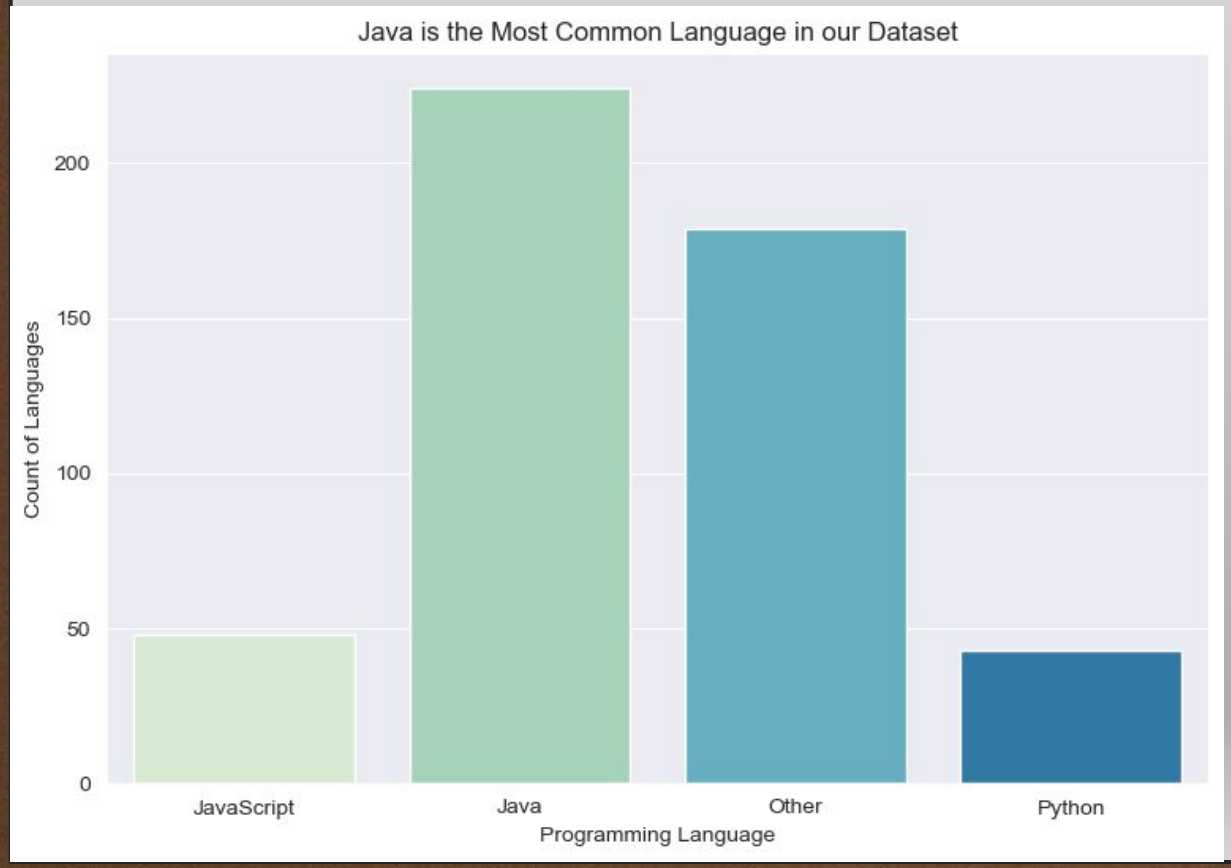
Acquisition | **Preparation** | Exploration | Modeling

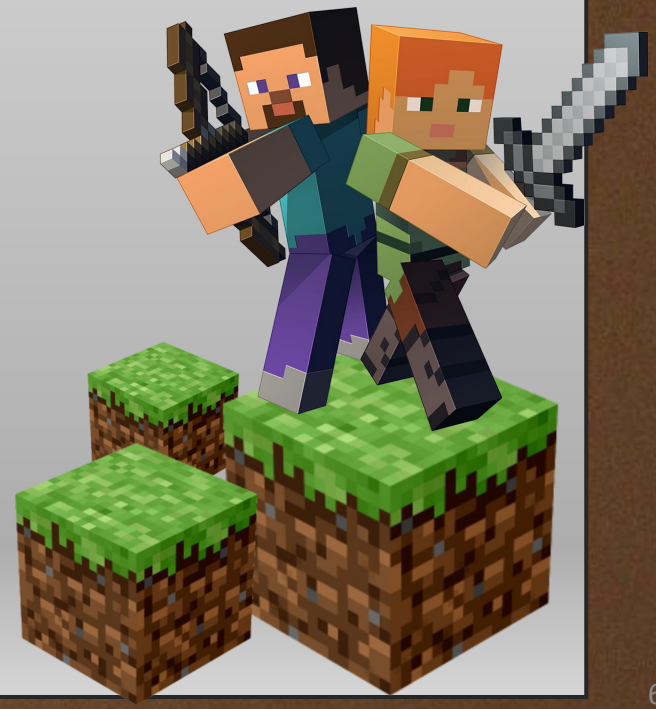
- Renamed columns to improve readability
- Removed common stopwords from values in readme_contents column
- Inspected integrity of data removing null values all rows where nulls existed
- Utilized Regex and string methods and functions to clean Repo readme text
- pred_readme_data() drops nulls, performs a basic clean, removes key stopwords,and lemmatizes the text

Data Science Pipeline

Acquisition | Preparation | Exploration | Modeling

Question 1: What are the top programming languages found in Minecraft related GitHub Repos?

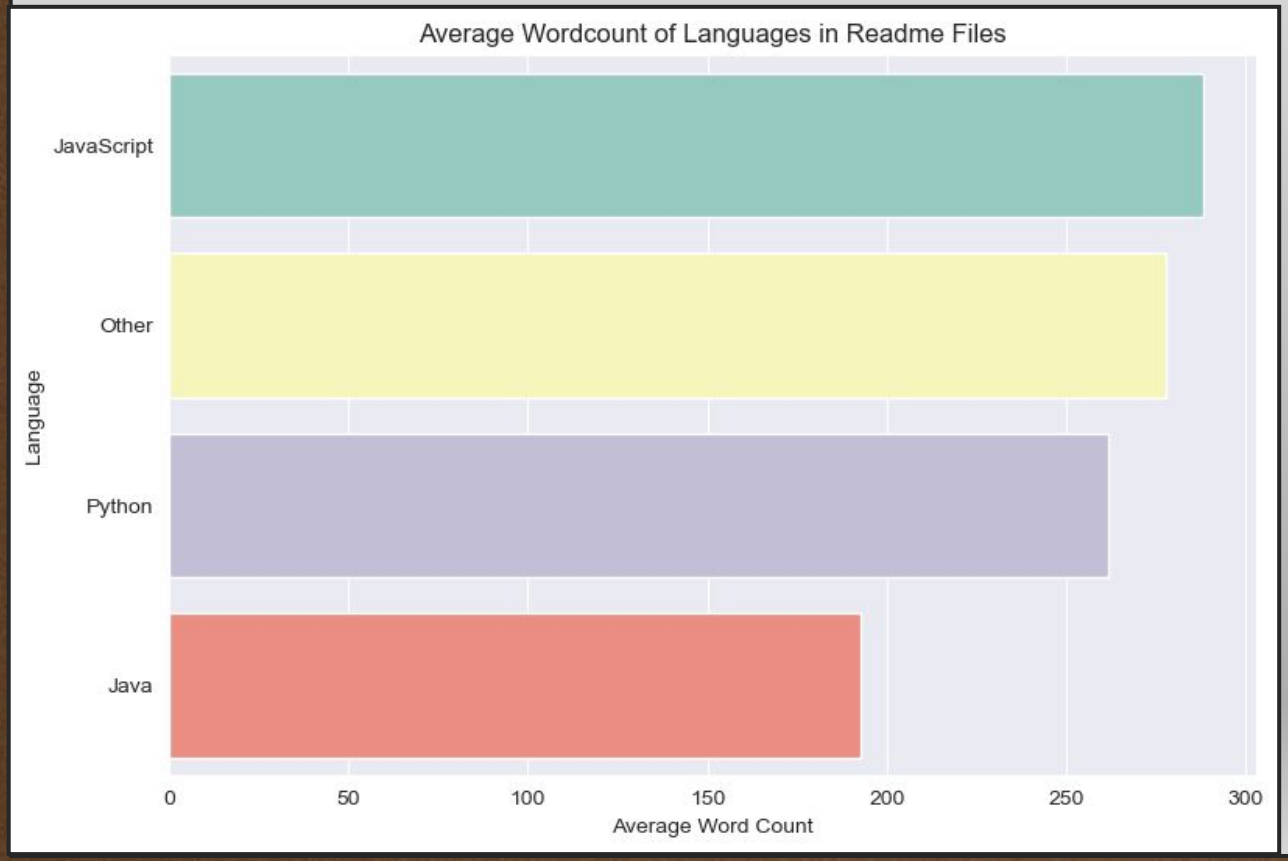




Data Science Pipeline

Acquisition | Preparation | **Exploration** | Modeling

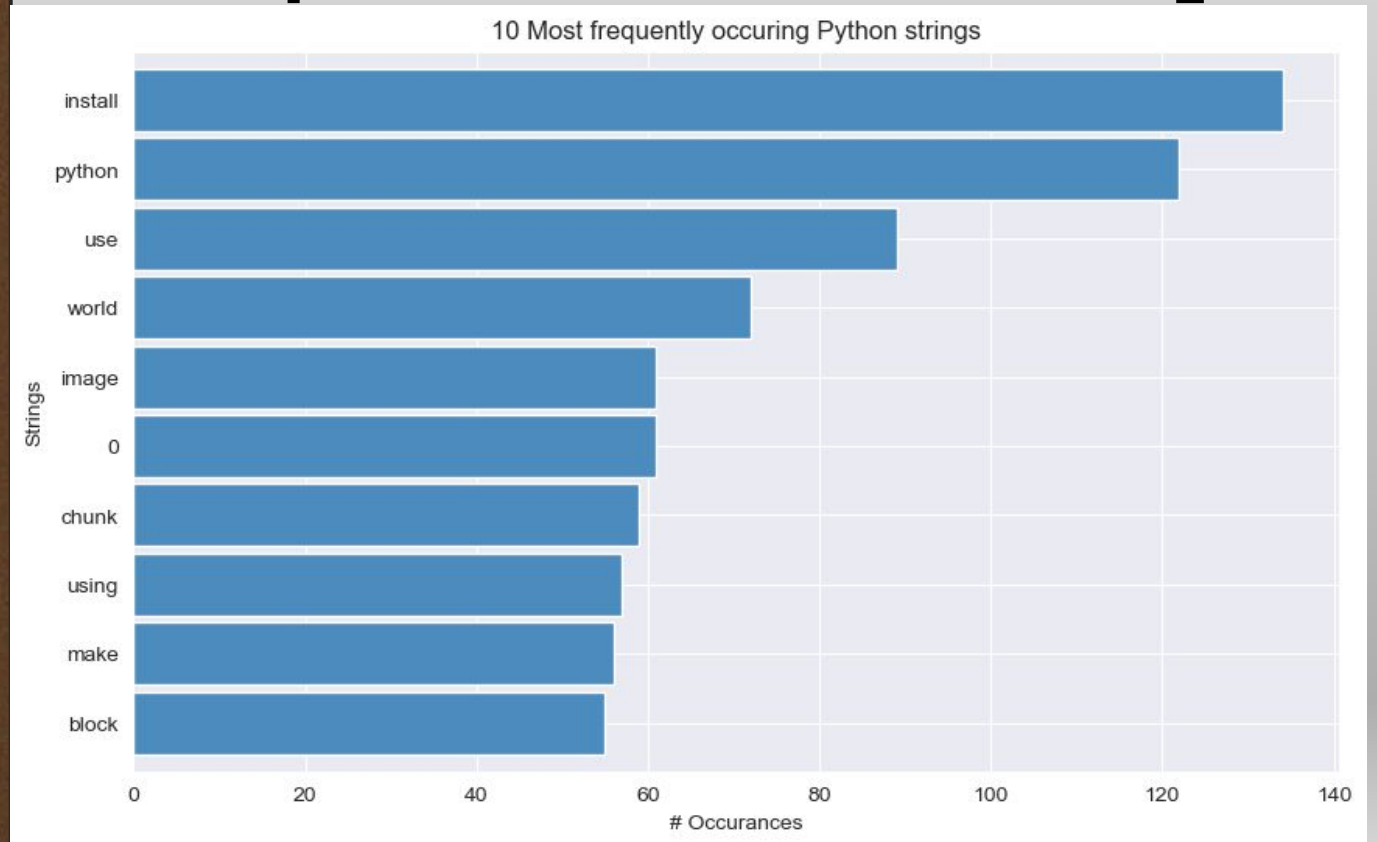
Question 2: What is the average word count of a Repo Readme file, based on their programming language?



Data Science Pipeline

Acquisition | Preparation | **Exploration** | Modeling

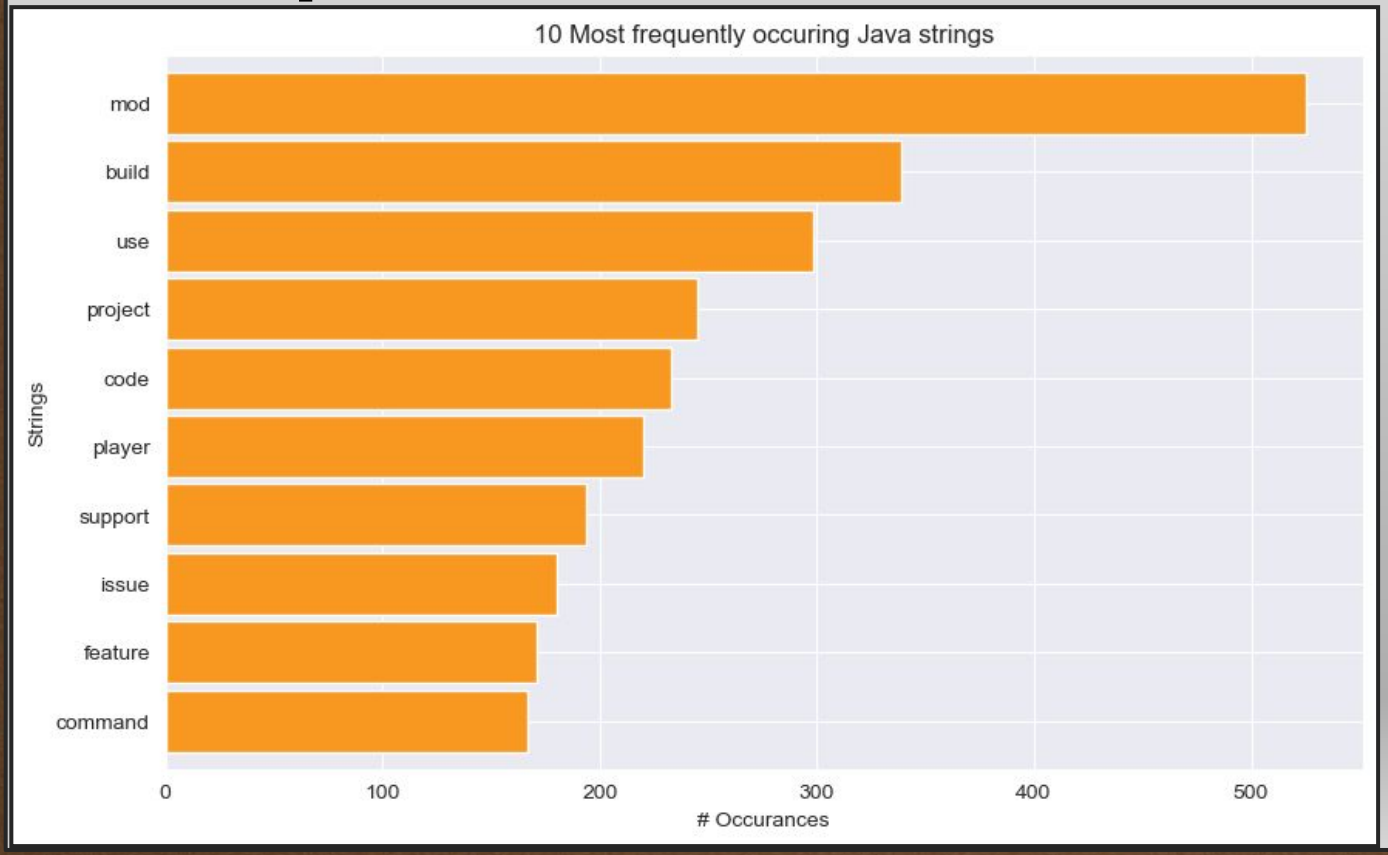
Question 3: What are the top 10 most frequent words found in Python Repos?



Data Science Pipeline

Acquisition | Preparation | Exploration | Modeling

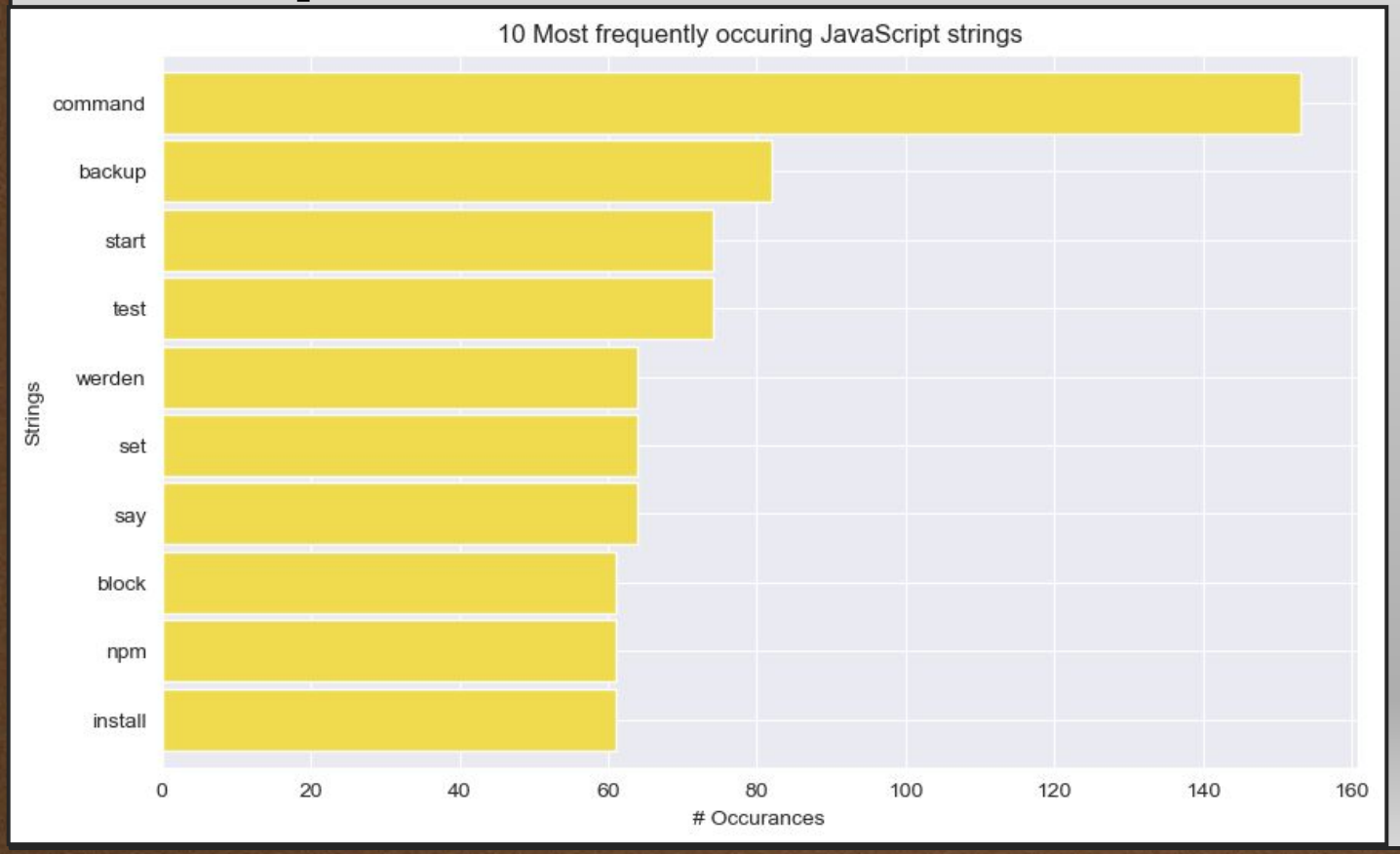
Question 4: What are the top 10 most frequent words found in Java Repos?



Data Science Pipeline

Acquisition | Preparation | **Exploration** | Modeling

Question 5: What are the top 10 most frequent words found in JavaScript?



Data Science Pipeline

Acquisition | Preparation | Exploration | **Modeling**

- We elected to utilize accuracy as the evaluation metric
- We developed three different models using different model types: (Naive Bayes, SKLearn Gradient Booster, XGBoost)
- The model that performs the best was evaluated on test data
- We utilized the mode of 'language' as the baseline (Java, 45.3%)

Data Science Pipeline

Acquisition | Preparation | Exploration | **Modeling**

- **Modeling Summary:**
- All models were overfit on the training data
- SKLearn Gradient Boost was chosen for test data
- This model performed with a 76% accuracy, a 30% improvement from the baseline

Executive Summary

Goal | Takeaways/ Conclusions | Recommendations | Next Steps

Project Goal:
Examine the GitHub Minecraft repo README files, build several predictive models that accurately predicts the coding language of the repository.

Executive Summary

Goal | Takeaways/ Conclusions | Recommendations | Next Steps

- GitHub Repos with different programming languages have significantly different features (Word count and unique words)
- Because ReadMe files are written in normal language, the accuracy of any model is limited
- Improved cleaning methods may increase model performance
- Count Vectorization (CV) in combination with ensemble classification is an effective modeling strategy for NLP/Text Classification problems

Executive Summary

Goal | Takeaways/ Conclusions | Recommendations | Next Steps

- Acquire longer Readme text files to feed into algorithm
- Narrow down parameters for classifications (more languages are more difficult to classify)
- Additional hyperparameter tuning may result in better model performance

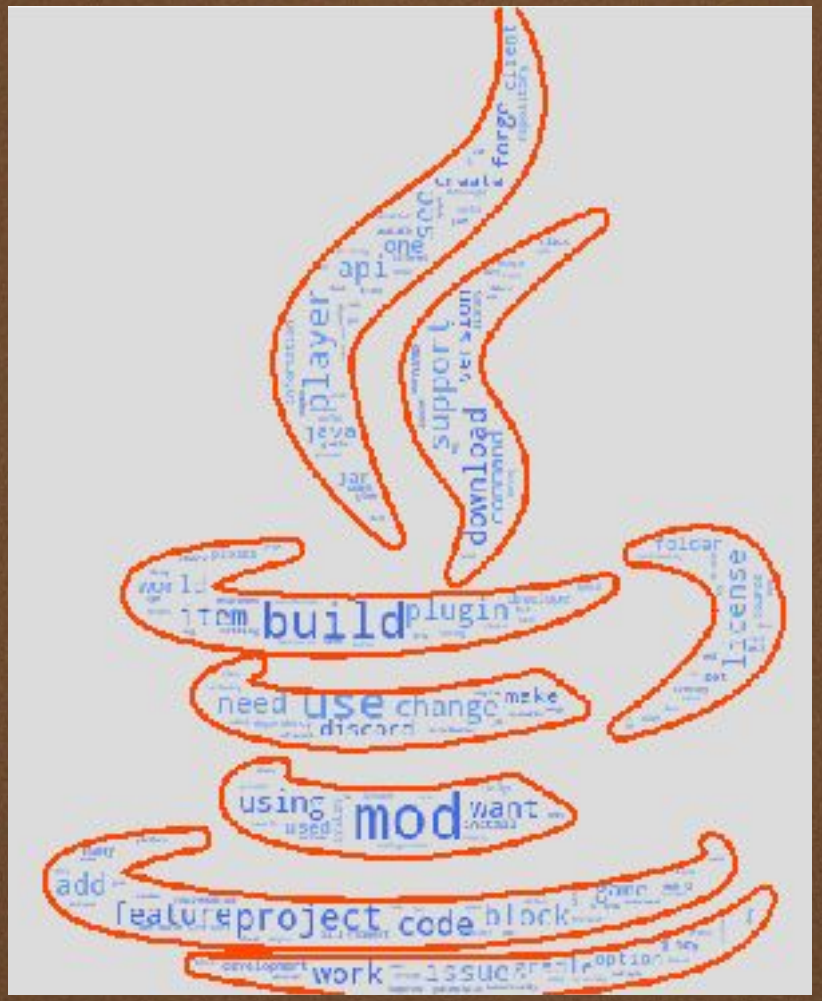
Executive Summary

Goal | Takeaways/ Conclusions | Recommendations | Next Steps

- Utilize statistical methods to identify additional stopwords
- Develop and test different model types for performance
- Find alternative methods for pulling repo data from GitHub

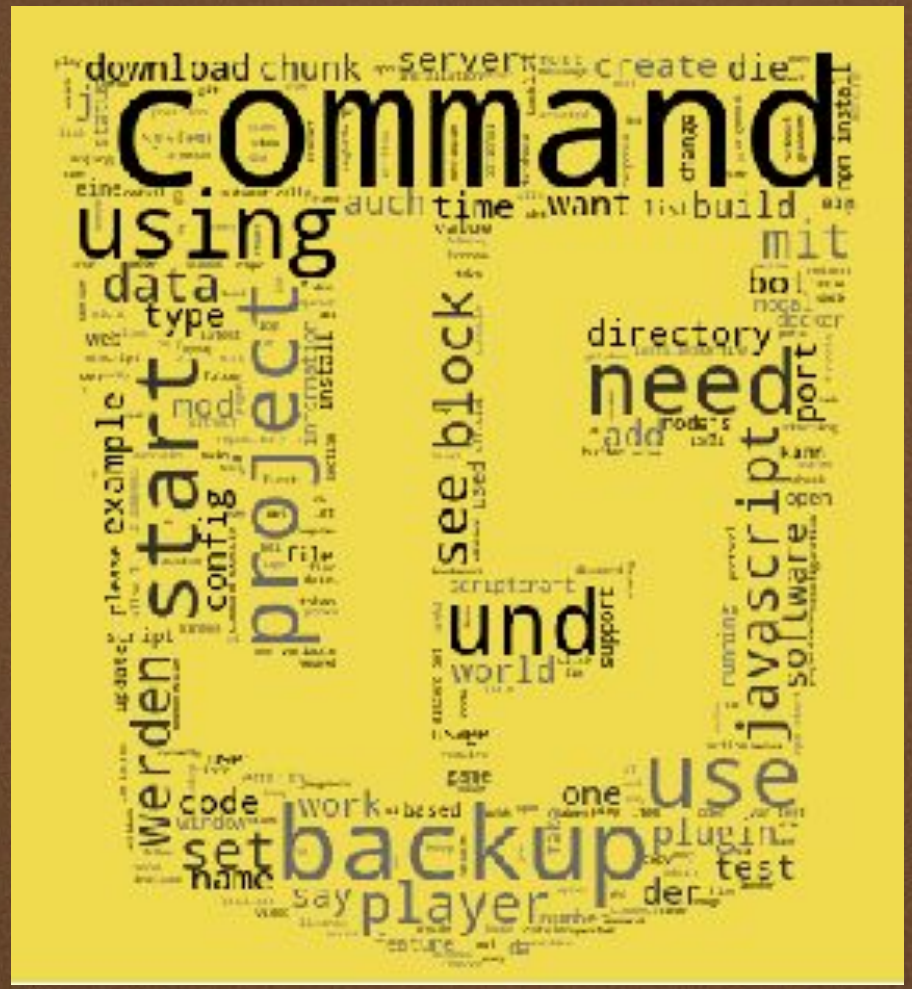
Word Clouds

Java | JavaScript | Python | Sword



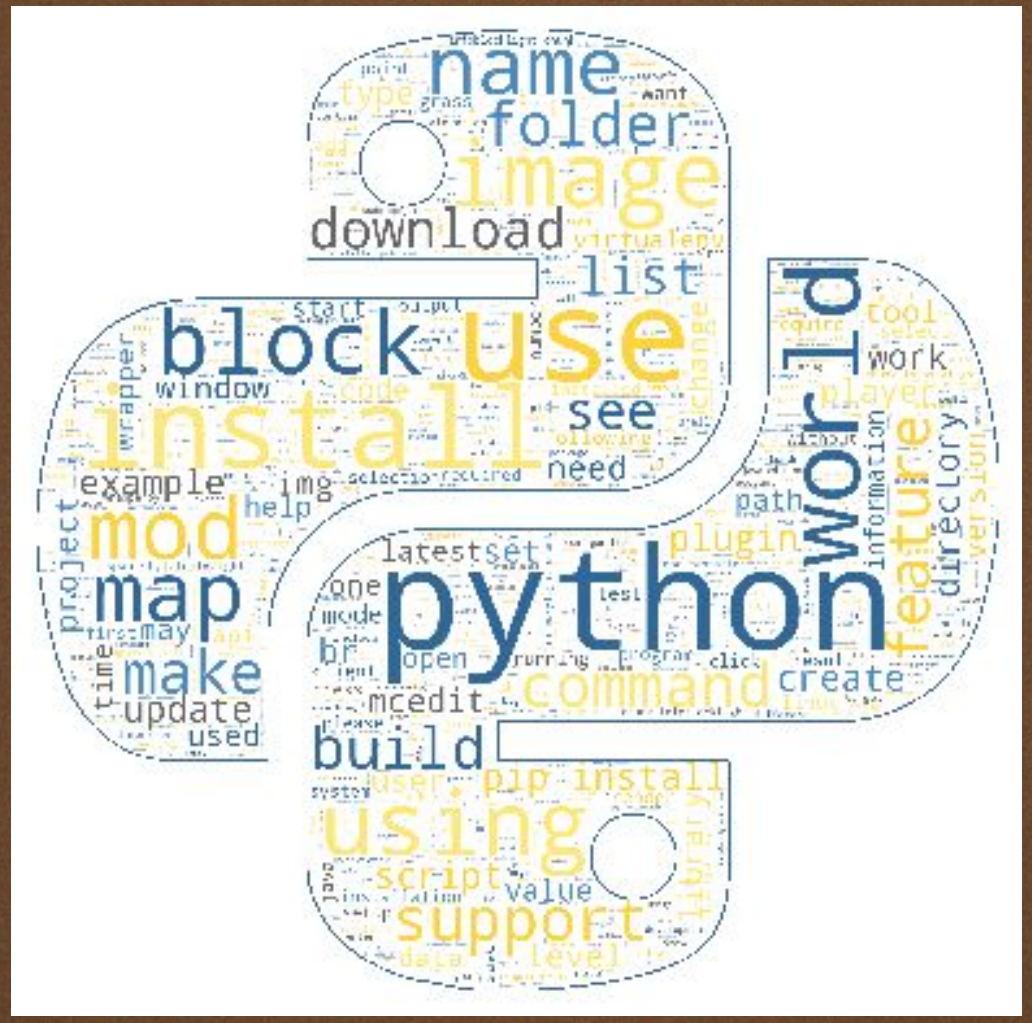
Word Clouds

Java | JavaScript | Python | Sword



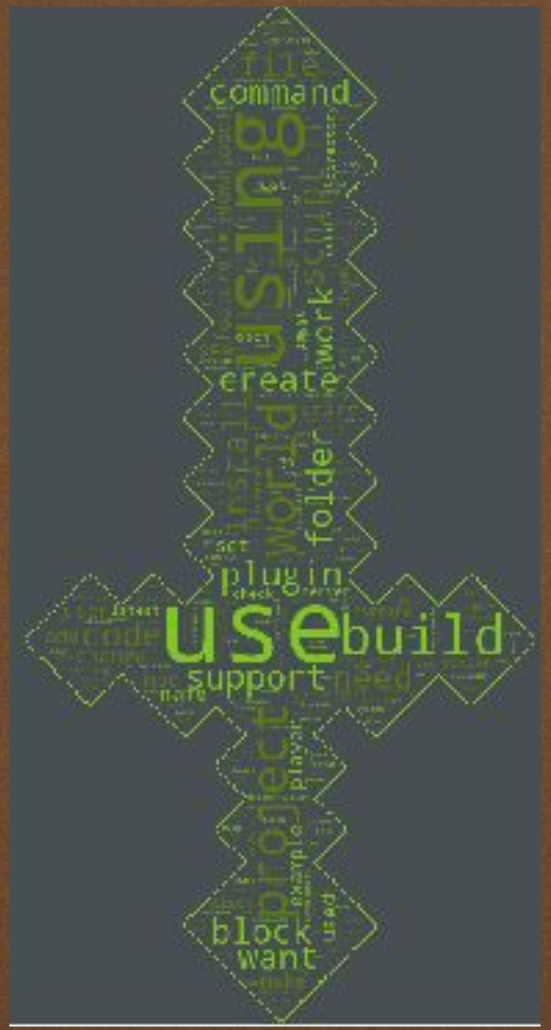
Word Clouds

Java | JavaScript | **Python** | Sword



Word Clouds

Java | JavaScript | Python | Sword








THANK YOU

Quit

Restart

PowerPoint Presentation

Slides courtesy of

Slide **Chef**