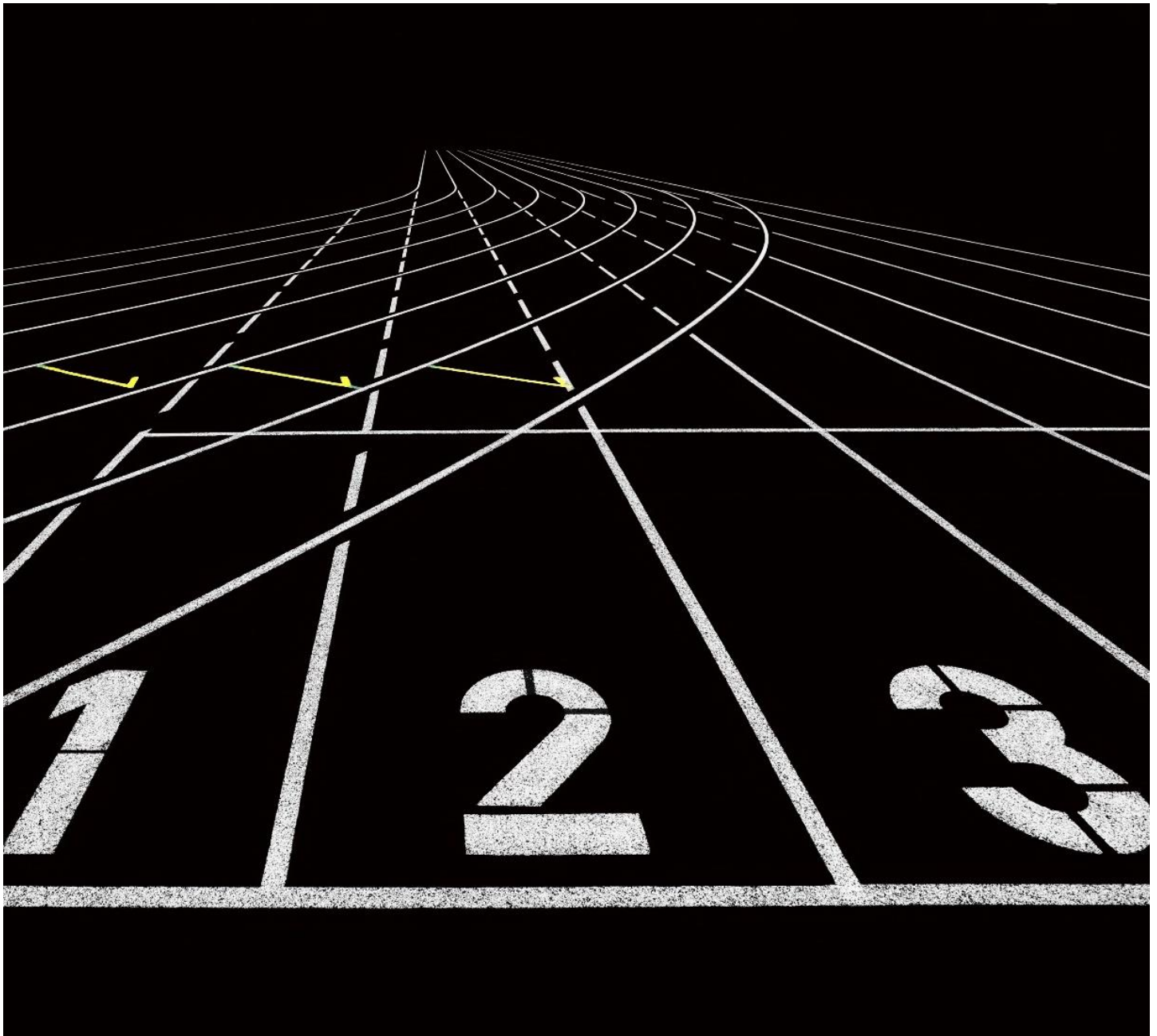


RECOMMENDING WEB ARTICLES

MAY 2022

MAHMOUD NAGY

pe.mn99@gmail.com



THOUGHT PROCESS

In order to extract features from the text we can use (Bag of Words, N_grams or TF-IDF). And to cluster our categories into subcategories, we can either use the title, or the article body. So, I examined combinations of the above.

I got a classifier accuracy of 85% using TF-IDF on the body and using the XGBoost Classifier.

For the Subcategory Clusters, I got different results, but saved those created using BOW on the titles, as I believe titles are more intuitive and representative in articles clustering, and I think BOW works best with titles than the TF-IDF method.

DATA CLEANING

Step 1: Check for NaNs

Step 2: Check for Duplicates

Step 3: Check short Titles

Step 4: Text Preprocessing

Step 5: Save The Cleaned DataFrame

Step 1: Check for NaNs

49 empty strings in the body column → replaced with 'missing'

Step 2: Check for Duplicates

- Dropped 20 duplicated rows
- Dropped rows where title is repeated, and the page does not exist or missing
- Dropped the "Learn More" Title
- Decided to keep other body and title duplicated in order not to lose information

Step 4: Text Preprocessing

- We can add a clean-text column in the data, then use string methods
- or we can extract features, then drop the unwanted columns

Preprocessing:

- Convert all text to lower case
- Remove underscores
- Remove words which contains same character more than twice (e.g. aaaaaaall)
- Remove Punctuations
- Remove Stop Words
- Remove Emojis
- Remove Non-ASCII Characters
- Remove all text starting with numbers
- Remove words with less than 3 characters

Note: I did not perform stemming for the cleaned text, and not sure if it will make a big difference. I preferred to leave words without stemming for better results interpretability

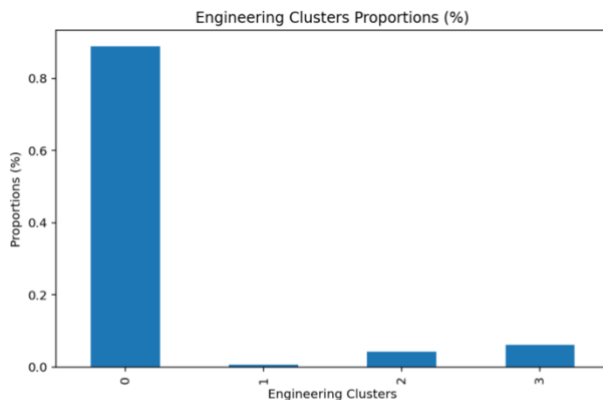
EXTRACT FEATURES FROM TEXT

In order to extract features from the text we can use (Bag of Words, N_grams or TF-IDF). I examined BOW and TF-IDF.

I think BOW works best with titles than TF-IDF and not sure if my intuition is right

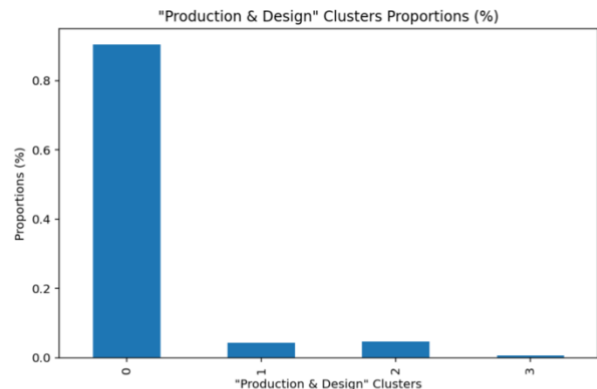
- As titles has way less words (but more representative) than the body,
- And repeating the word in many titles should show its importance for the clustering,
- This shouldn't be penalized and treated like "stop words" as in TF-IDF.

Also, I noticed that most of the articles fall into one cluster, when using **TF-IDF** with **titles**.



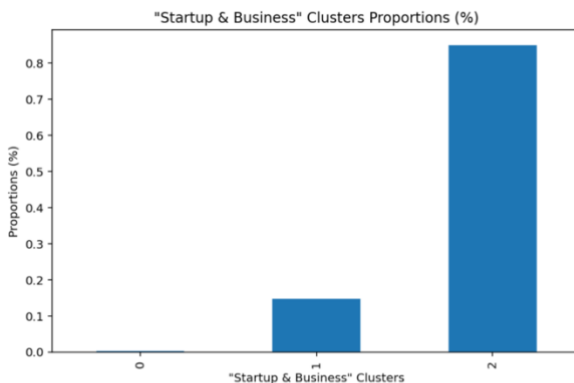
NOTE:

- The majority of the engineering articles falls into cluster (0)



NOTE:

- The majority of the Product & Design articles falls into cluster (0).



NOTE:

- The majority of the Startup & Business articles falls into one cluster (2).

BUILD A SUPERVISED LEARNING MODEL

I got a classifier accuracy of 85% using TF-IDF on the body and using the XGBoost Classifier.

I experimented with some of the ensemble methods, as they are powerful in getting high accuracy from simple models. RandomForest is much faster than XGBoost. *We can also experiment with other models too like SVC and Naïve Bayes.*

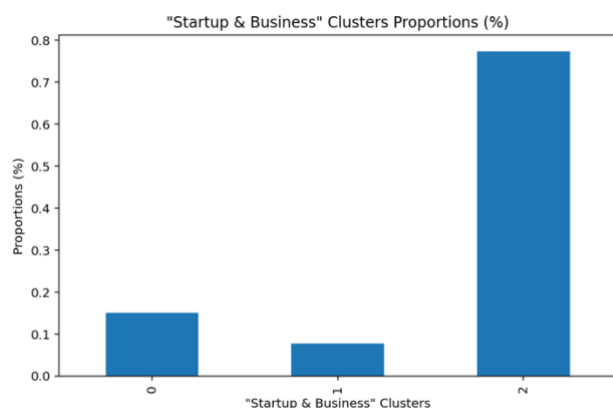
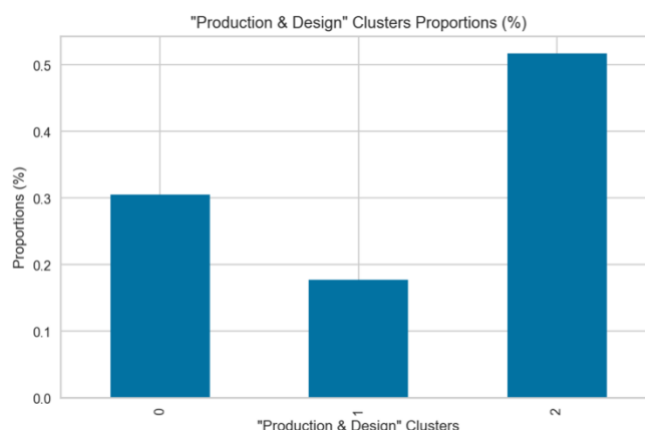
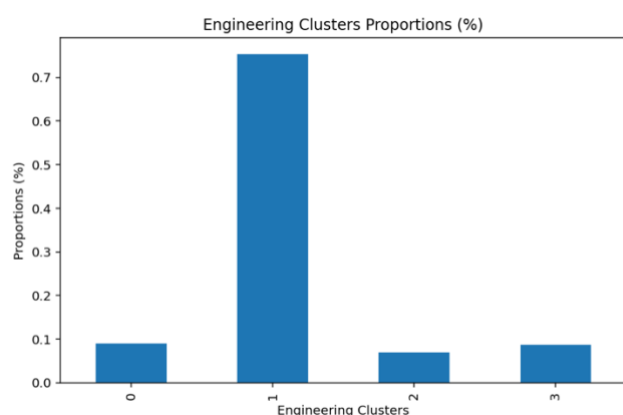
CLUSTERING

For the Subcategory Clusters, I got different results, but saved those created using BOW on the titles, as I believe titles are more intuitive and representative in articles clustering, and I think BOW works best with titles than the TF-IDF method.

I only experimented with KMeans, but I think we need to experiment with more algorithms like: Hierarchical clustering, and DBSCAN.

I used the **elbow method** along with **silhouette coefficient** to pick the number of clusters to be used. I know silhouette is for validation, but we did not have a clear elbow. For Engineering, Silhouette Score suggests k=3, but I used 4 clusters as given (prior knowledge).

Clustering results, using BOW with titles:



Clustering results, using TF-IDF with body:

