Motivation:

- Figure out what content I spend time watching on Youtube- what do I use youtube for? eg. productive, class stuff, entertainment, news?
- Learn new tools like beautiful soup and NLTK

Challenges:

- With BeautifulSoup: Annoying to deal with, needed to convert soup into list and apply list comprehension, and convert back into soup etc. to make a filtered version of soup with the data we want (without ads, etc.)
- We don’t have the category labels for each video record. We need to find someway to attribute categories to each record, then we can analyze how the trends change over time
    - API approach to labeling our video categories
        - Costly. We can just try to train a model based existing datasets instead

NLTK for title name processing

- The way we preprocess the text
    - TF-IDF takes into account the frequency of the words as well as the uniqueness of words across documents
    - Bag of Words only considers frequency of words
    - LDA is good at handling large sets of text data and can uncover interesting topic patterns without predefined categories.
        - Gensim vs Sklearn
            - SKlearn is computationally a lot faster
- Tokenizing was a hassle as well. Sometimes only got emojis, punctuations etc.

Clustering approach

- K-means
    - Large inertia since we’re dealing with noisy text
    - Choosing the parameters
        - Increasing n_init value to 20 because we’re dealing with a larger dataset, init = ‘k-means++’ worked better in choosing an initial centroid
            - After increasing n_init, we got a monotone graph again whereas before when using the default n_init = 10 the graph would spike up at points
- Hierarchical
    - Ward linkage seems to be best for noisy data. And titles have roughly the same length.

<u>Reflection</u>

After having narrowed down to k many clusters, I still had to take a sample of each cluster, look at the titles, and try to interpret the category

- One issue is that we’ll have a category like “shorts” which is not an entertainment category

- Also in the distribution of cluster sizes, one might be abnormally large and nondescript

Idea: I could also implement LDA topic recognition here

Reflections on approaches

- K-means is giving me very mediocre categories, when running a second iteration of k-means, it’s showing that it’s no longer able to do a good job beyond splitting into common keywords (not helpful because we’re not getting the semantics)
- So perhaps I could train a prediction model on an existing youtube dataset instead

Finding a good dataset is hard! Some lack the breadth of categories, and desired size

The idea for a solution:
Combine all the datasets- US videos, FR videos, GB videos, (all the ones with relevant languages that I would watch). Should delete repeat video title entries (don’t want to overtrain model on specific titles)

- Training wasn’t great since some categories were much heavier in the dataset, so I had to balance it out more
- REDUCING the number of categories. Precision increased a lot

Ideas for next projects:

- Analyzing school meal plan usage- which dining halls do I frequent and when? Which cafes and when?
- Apple Health sleep data