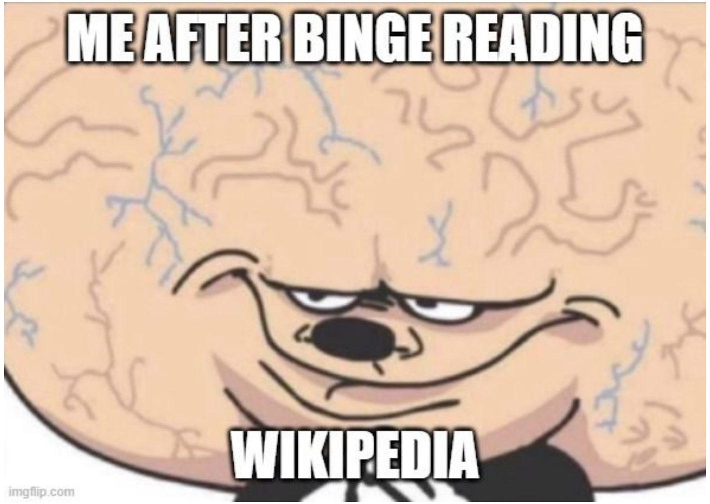
Wikipedia Clustering with K-means

Kyle Wong, Maria Lee, Jim Xu

MIDS 207

The Problem with Browsing Wikipedia

Have you ever gotten so tired of social media that you started reading Wikipedia pages instead? How do you find an enjoyable Wikipedia page?

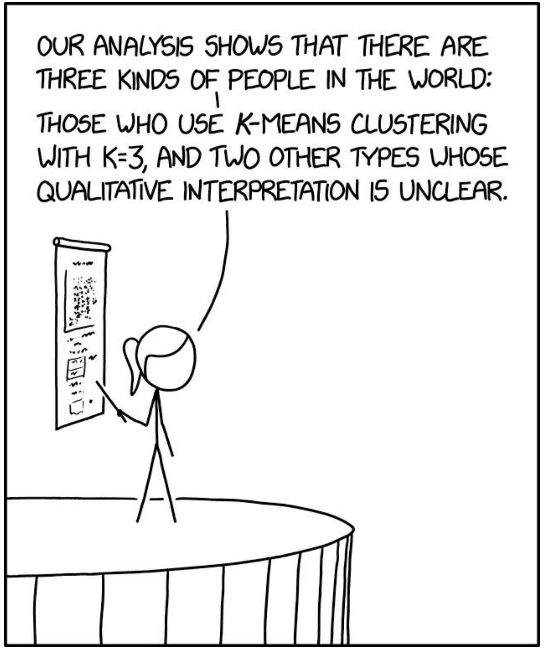


While Wikipedia already recommends and upsells articles to you, there are still a few remaining user problems that can be tackled:

- How can we find relevant, enjoyable articles without being constrained by topics?
- Is there a way to categorize articles based on similarities in semantics and content style?

We Come with a Solution

Goal: Create a clustering model to sort articles into custom categories. Categories are cross-topics. Each article would be sorted into only one category.



Data Set

We use 2 datasets in our project: Figshare and web-scraped Wikipedia articles

- The Figshare set contains **64 categorical labels** of articles:
  - Categories were created by a team of researchers
  - **English Wikipedia (EN)**
- Scraped article text from English Wikipedia for the Figshare dataset

Web Scraping

- Filtering out only English Wikipedia articles from Figshare’s full dataset
    - (6.236.637, 68)
- Added url of each article using unique page id
- Reduced the dataset based on category counts - 1.47% of original counts
    - (91.608, 70)
- Used ‘BeautifulSoup’ to extract 300 words of text for the articles via url
- Utilized ‘ThreadPoolExecutor’ to parallelize the scraping to 10 threads

Methodology – Data Processing for Modeling

- Prepared text by removing non-English text and unconventional punctuation marks for data uniformity
- Tokenized the text and develop a word index to map each unique token to a numerical identifier
- Passed tokenized text through an embedding layer to vectorize semantic meaning
- Ran K-means on a variety of cluster sizes and analyzed outcomes

Methodology – Determining Clusters

We have simulated our model on 4 cluster sizes: 64, 31, 8, and 4. These clusters correspond to the natural categorization of Wikipedia articles and optimal cluster size yielded from the Elbow Method:

Low-Level Categories (64):

['Culture.Biography.Biography*', 'Culture.Sports', 'STEM.Biology', 'Geography.Regions.Americas.North_America', 'Culture.Media.Media*', 'STEM.STEM*', …]

Sub-Level Categories (31):

['Libraries_&_Information', 'Visual_arts', 'Technology', 'Internet_culture', 'Business_and_economics', 'Education', 'Physics', 'Biology', 'Computing', '...]

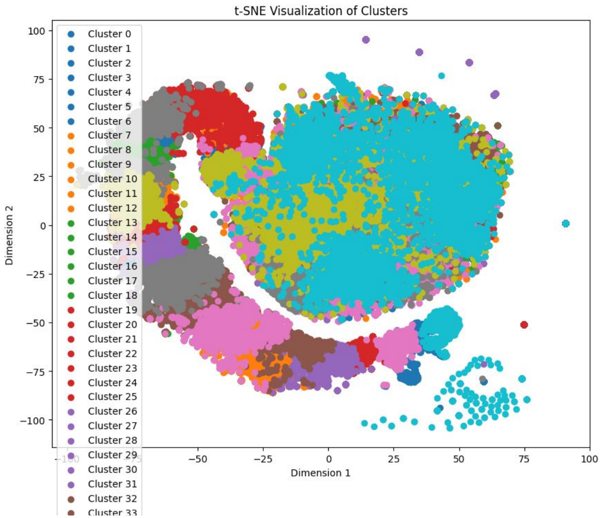
Top-Level Categories (4):

['Culture', 'History_and_Society', 'Geography', 'STEM']

Elbow Method (8)

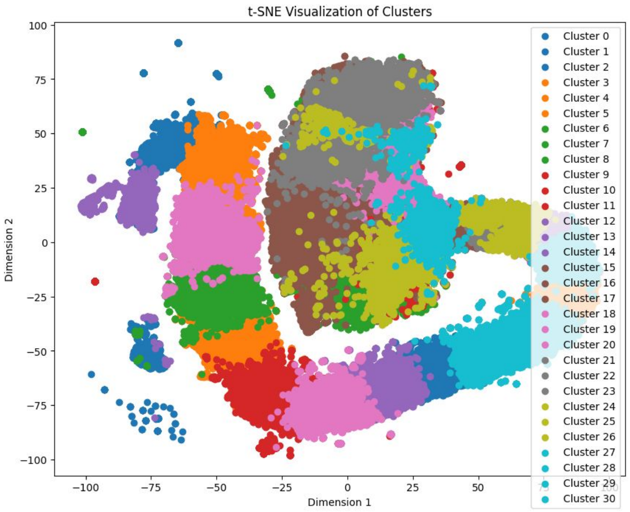
Modeling K-Means: 64 and 31 Clusters without PCA

64



Inertia = 65.52

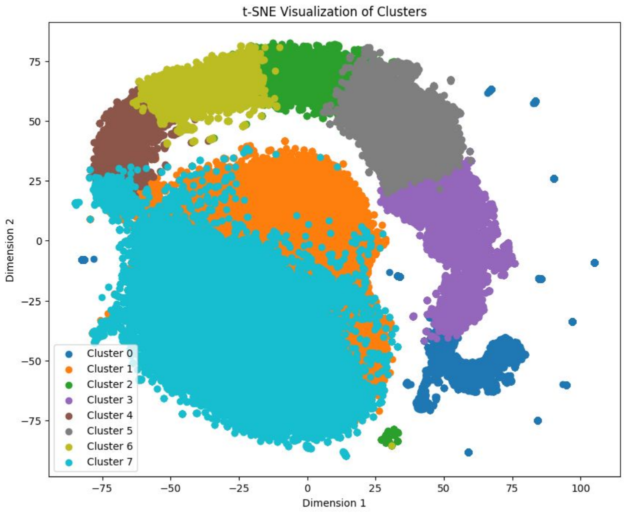
31



Inertia = 68.42

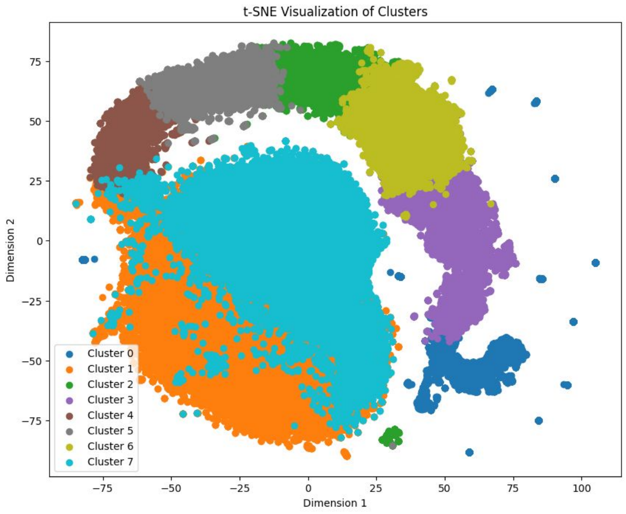
Modeling K-Means: 8 Clusters

No PCA



Inertia = 82.16

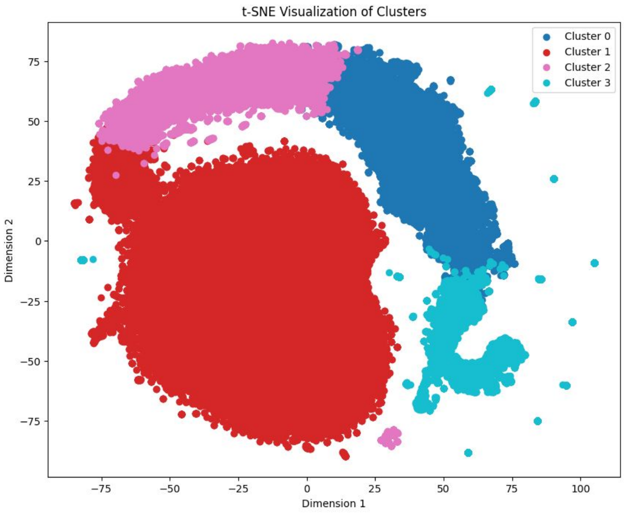
PCA = 8



Inertia = 28.14

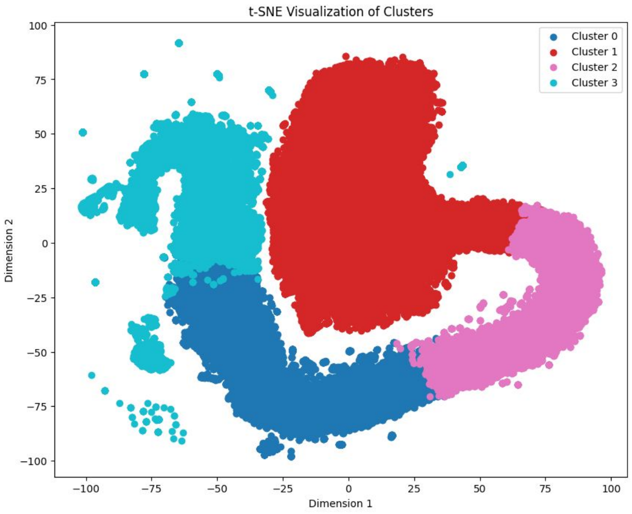
Modeling K-Means: 4 Clusters

No PCA



Inertia = 113.95

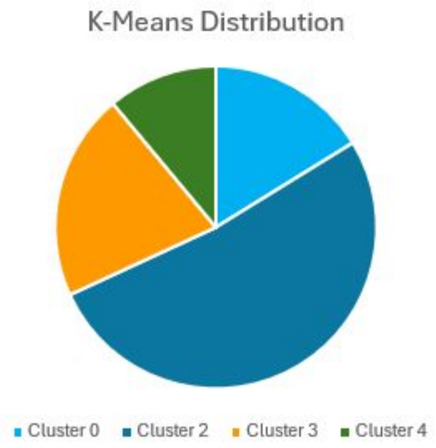
PCA = 8

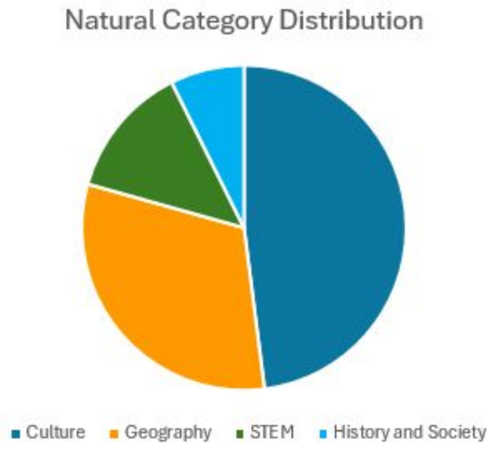


Inertia = 59.97

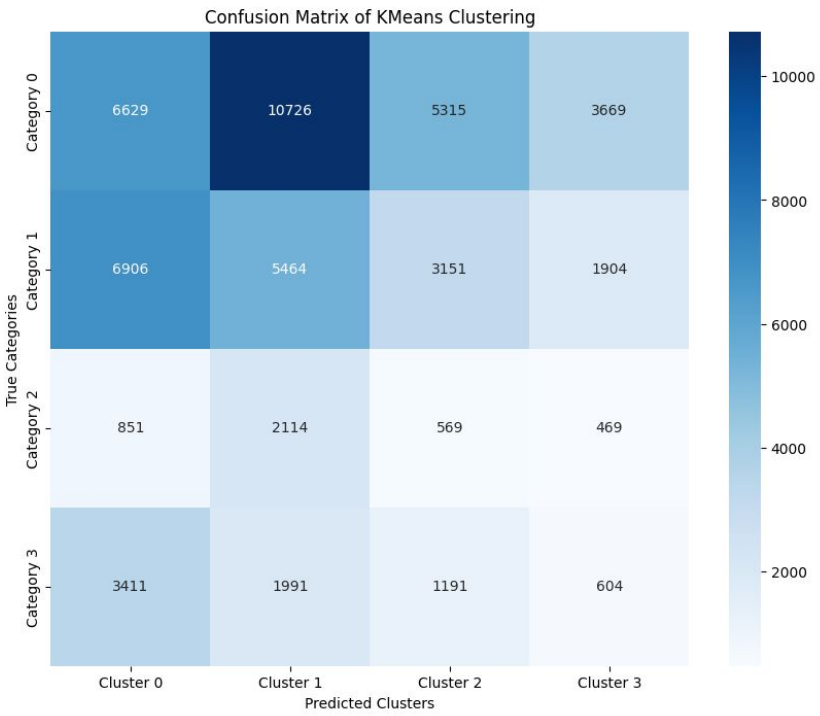
Choosing the "Best" Model

- Statistical winner: 8 clusters with PCA
  - May not be clear distinction between articles of adjacent groupings
- Contextual winner: 4 clusters with PCA
  - Should be greater distinction between articles
  - Allows for easier comparison against natural categorization by 4 top-level categories





Modeling K-Means: Confusion Matrix (4 Clusters)

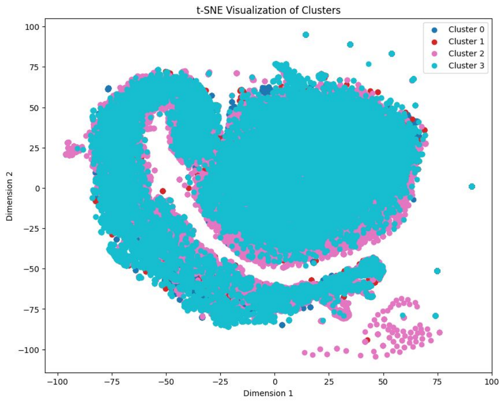


- Category 0 (Culture) is most confused with cluster 1
- Category 1 (Geography) is most confused with cluster 0
- Category 2 (History_Society) is most confused with cluster 1
- Category 3 (STEM) is most confused with cluster 0

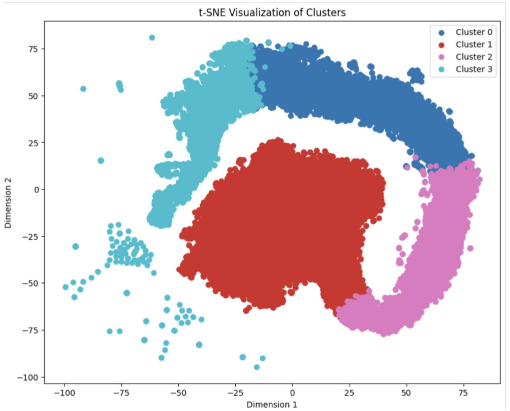
Model confuses articles as cluster 1 the most and then cluster 0.

Labeled Category Baseline Comparison

Baseline



K-Means



How do pre-defined categories look in our 2-D dimensional embedding space?

['Culture', 'History_and_Society', 'Geography', 'STEM']

Performance Evaluation *w.r.t. labeled categories*

- Accuracy: 24.13%
- Precision: 19.70%
- Recall: 19.78%
- F1 Score: 18.93%

Practically…

- Category is not being extracted and used to group with K-Means
- Articles across topics share similar semantics and styles
- Articles within topics may contain drastically different writing styles and word choices

Modeling – K-means Common Word Groupings

Group 0:

unable: 6507

north: 6464

help: 5749

football: 1989

south: 1661

km: 1615

film: 1491

Group 1:

article: 6700

stub: 6635

help: 6060

film: 5861

also: 5747

first: 5214

wikipedia: 5019

Group 2:

wikipedia: 2289

article: 2288

stub: 1978

born: 1831

played: 1475

first: 1459

school: 1387

also: 1325

Group 3:

km: 4757

wikipedia: 4500

help: 3392

south: 3090

article: 2910

also: 2735

stub: 2544

football: 2096

Conclusions

- It appears that it is very difficult to group Wikipedia articles by semantics and content style as there are multiple contributors (and thus writing styles) to articles.
- While we appreciate tl;dr / executive summary sections in articles, our model cannot strongly group articles together based on semantics–thus, tl;dr sections may be less useful statistically in determining the overall takeaway and quality of articles than via observation.
- K-means may not be the best algorithm for clustering complex data.

Continued Work

- Testing different algorithms for categorization of Wikipedia articles instead of K-means
- Utilize a Wikipedia dataset on Kaggle with quality rankings to determine the quality of the wikipedia articles based on scraped text
- Using model categorization to personalize article recommendations to readers based on their interaction

Thank you!

Questions?