Clustering Federal Register Comments for Efficient, Manual Review

Brandon Kopp

Bureau of Labor Statistics

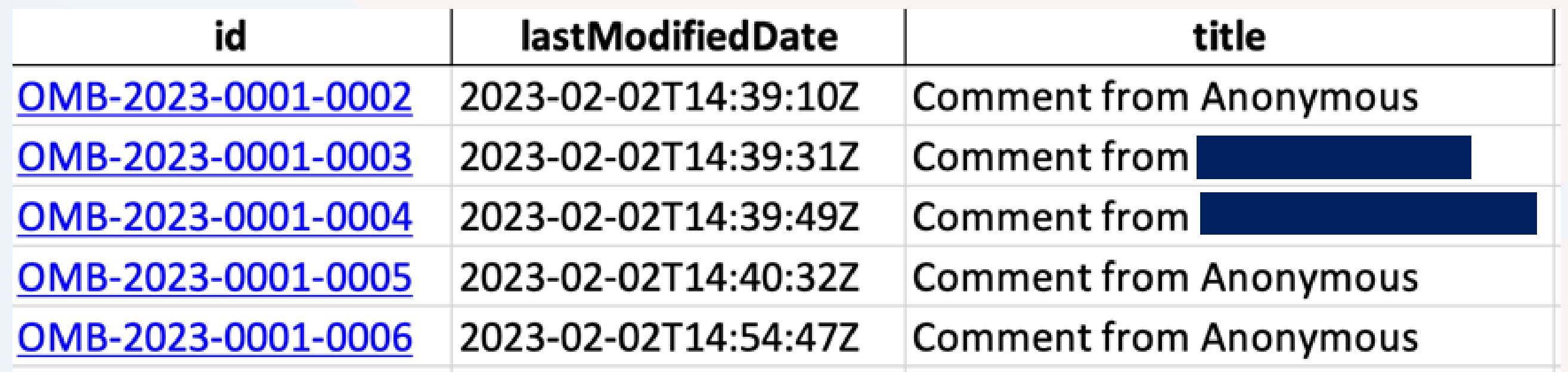
Problem

- In 2023, an interagency team was tasked with reviewing over **20,000 public comments** received in response to a Federal Register Notice on possible revisions to race and ethnicity measurement standards
- To aid in manual review, a data pipeline was set up in which:
  - Comments were retrieved daily during the comment period through the **regulations.gov API**
  - Letter writing campaigns (exact and near duplicate comments) and unique comments were identified using **text embeddings** and **clustering techniques**
  - Information was shared with reviewers and stakeholders through a dashboard and Excel spreadsheet
- While review time was not measured, reviewers credited the clustering information with speeding up the process of labeling comments and quantifying themes in the comments

Retrieving Text Data

The public post their comments to FRNs using regulations.gov. They can submit text comments through a form and/or attach documents (PDF, Word, Images).

Most manual comment review work happens in spreadsheets. The admin panel on regulations.gov outputs a spreadsheet with minimal information with links to online comments which can be burdensome for reviewers.

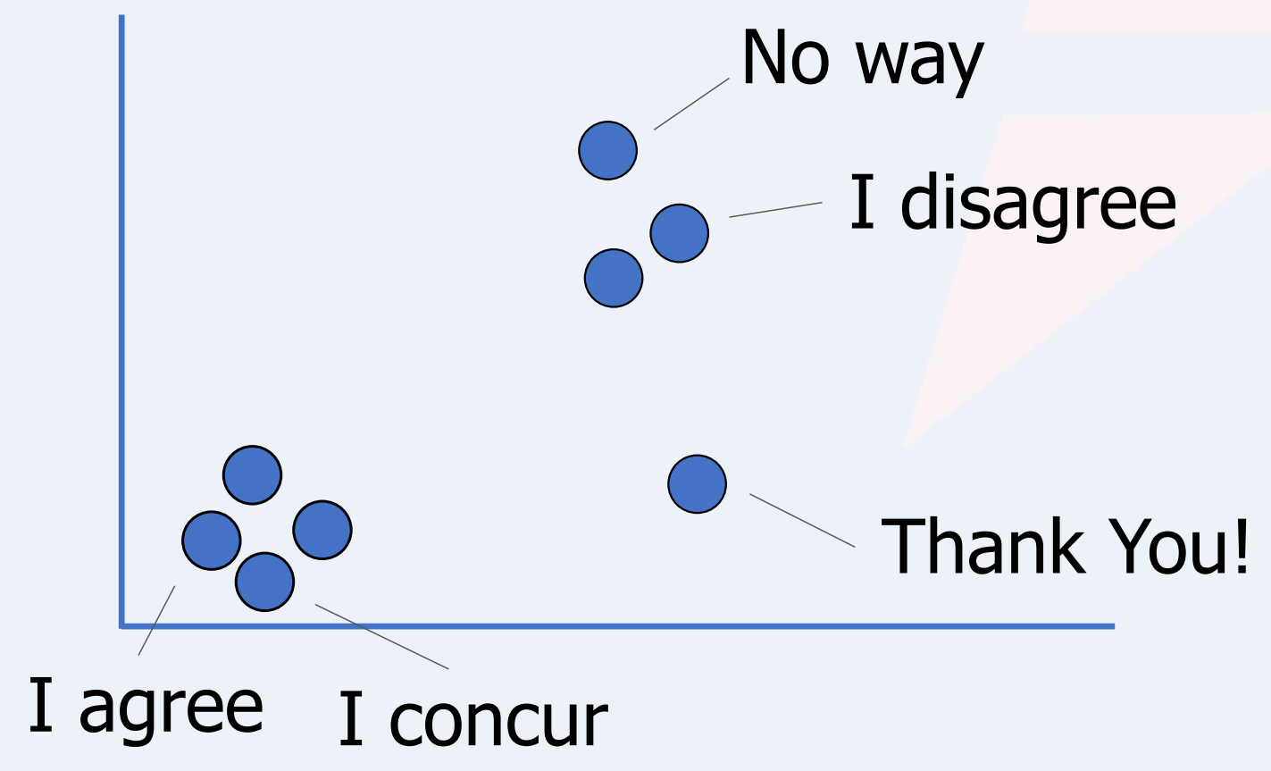


id	lastModifiedDate	title
OMB-2023-0001-0002	2023-02-02T14:39:10Z	Comment from Anonymous
OMB-2023-0001-0003	2023-02-02T14:39:31Z	Comment from [redacted]
OMB-2023-0001-0004	2023-02-02T14:39:49Z	Comment from [redacted]
OMB-2023-0001-0005	2023-02-02T14:40:32Z	Comment from Anonymous
OMB-2023-0001-0006	2023-02-02T14:54:47Z	Comment from Anonymous

I automated the process of downloading comment metadata, including the full text of comments, from the regulations.gov API using the `requests` package. New comments were downloaded daily and stored in a local database using the `pandas` and `sqlite3` packages for later processing.

Text Embeddings for Comments

I agree	0.63	0.02	...	0.15	0.98
I concur	0.59	0.10	...	0.19	0.92
I disagree	0.13	0.04	...	0.14	0.95



Text embeddings encode the semantic meaning of a provided text into a vector of numbers (in this case a 768-dimensional vector). Similar comments should be close to one another.

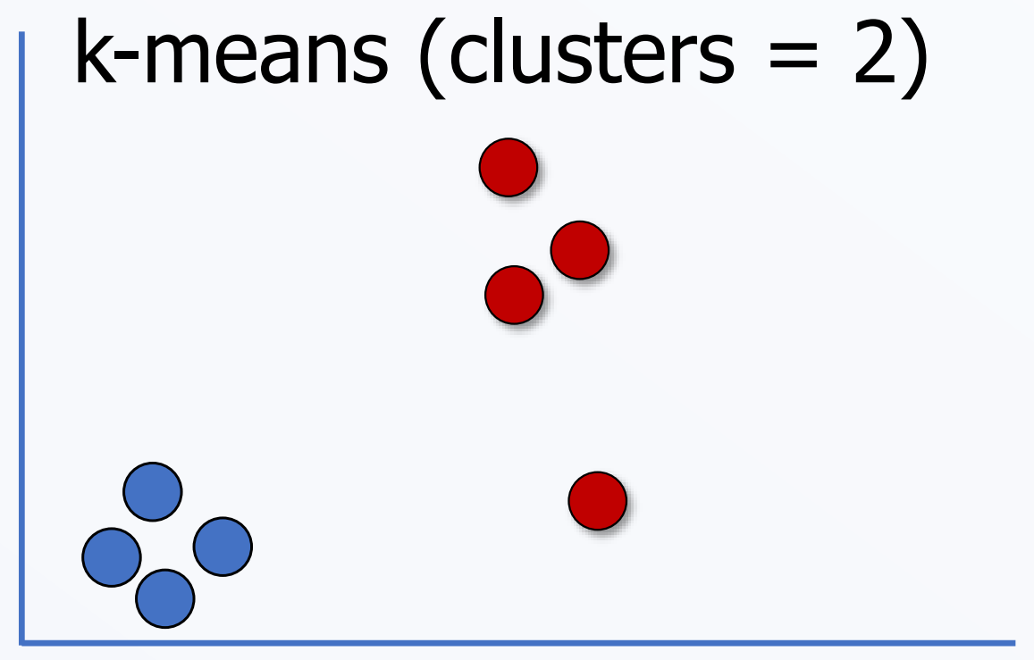
For this project, I used a pre-trained model trained on large amounts of text (a variant of the BERT model) through the `sentence-transformers` package in Python.

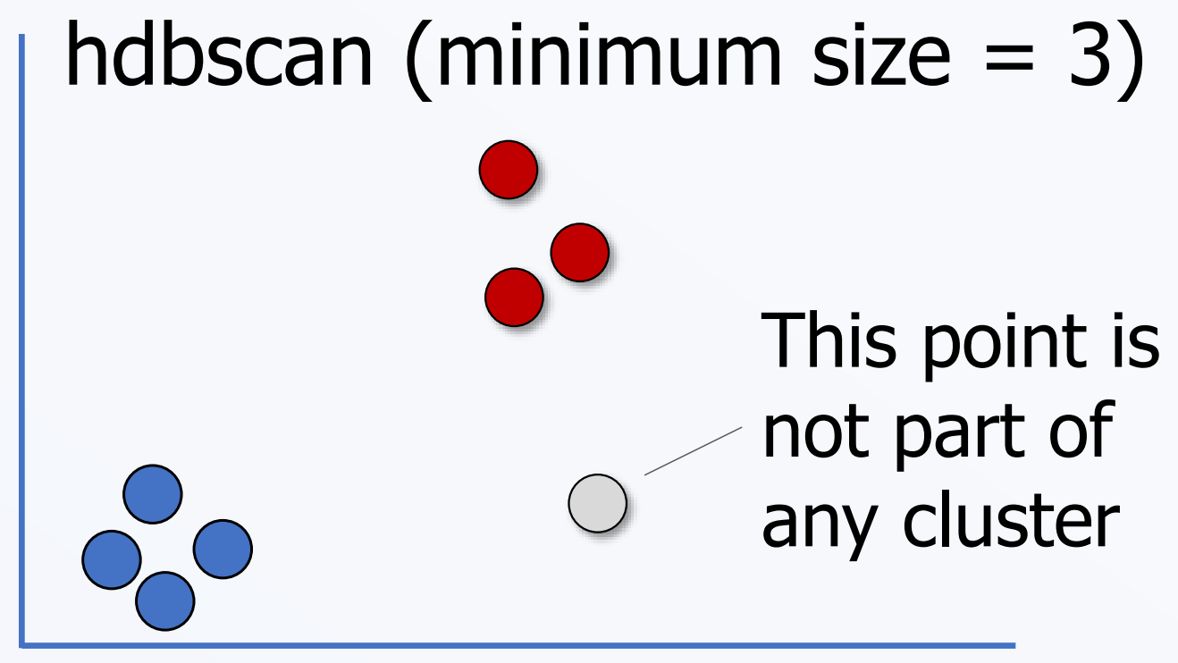
Clustering

Clustering methods group together items close to one another in multi-dimensional space.

Some methods require pre-defining the number of clusters (k-means) and some define the number of clusters algorithmically (hdbscan). hdbscan also allows for the possibility that some comments may not fit neatly into a cluster.

I clustered comment embeddings and provided outputs of both methods to reviewers using the `scikit-learn` and `hdbscan` packages.




hdbscan (minimum size = 3)
This point is not part of any cluster

Visualization (optional)

I agree	0.63	0.02	...	0.15	0.98
I concur	0.59	0.10	...	0.19	0.92
I disagree	0.13	0.04	...	0.14	0.95

768 dim

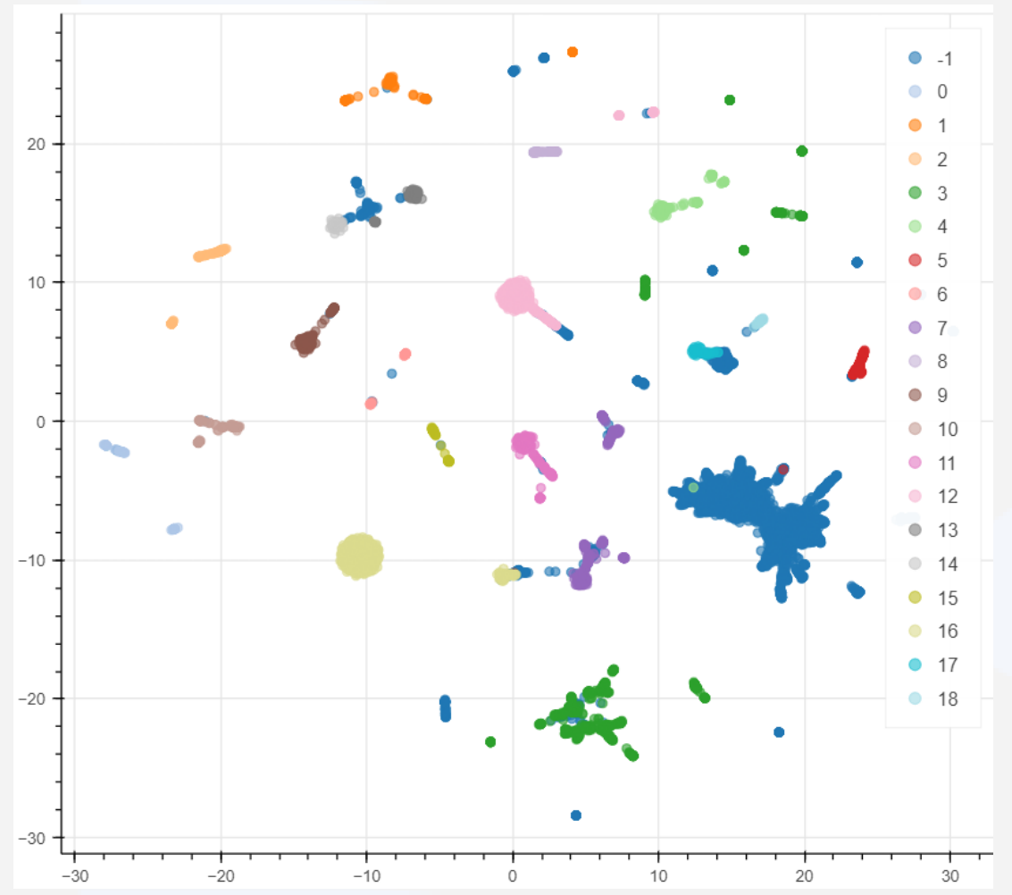
2 dim

I agree	0.15	0.02
I concur	0.20	0.10
I disagree	0.68	0.82

Dimensionality Reduction: When visualizing clusters, we need to reduce dimensionality to something our mind can comprehend. In this case, I reduced the 768-dimensional comment vectors down to 2 dimensions.

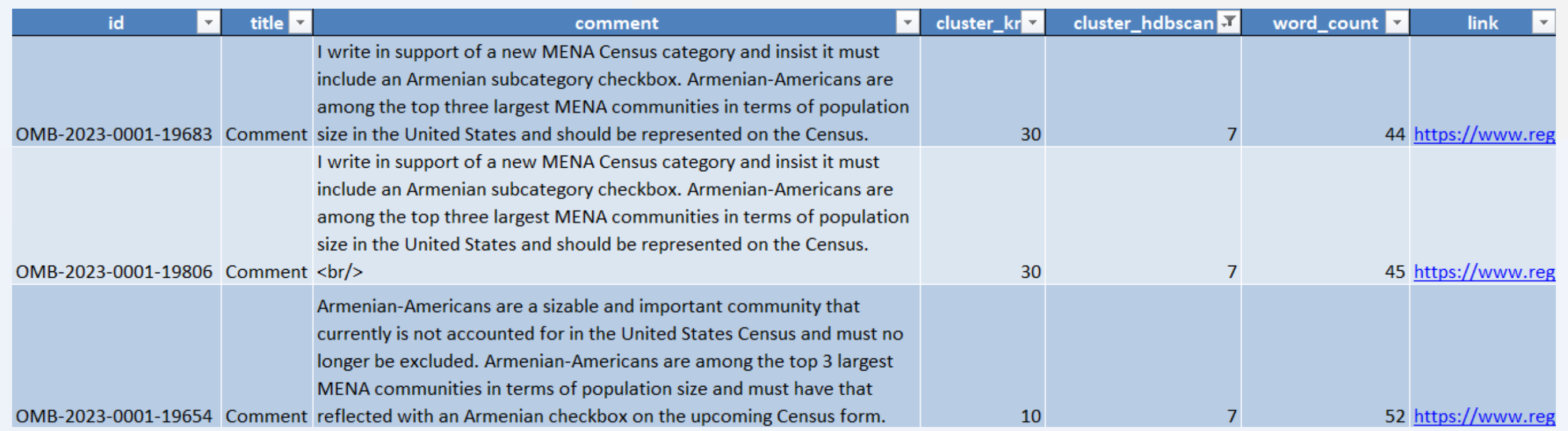
There are a number of algorithms for this (PCA, TSNE, UMAP, PACMAP). In this project, I used the `pacmap` package.

Visualization: Visualizing clusters can be useful for understanding how well the clusters represent the underlying data. Tight, well-spaced clusters are preferred. The use of an interactive visualization package like Python's `bokeh` can allow you to explore clusters by hovering over points and seeing their contents.



Final Products

An updated spreadsheet was provided to reviewers and stakeholders daily during the comment period through an R `shiny` dashboard. This included full comment text and clusters that could be used for sorting and filtering.

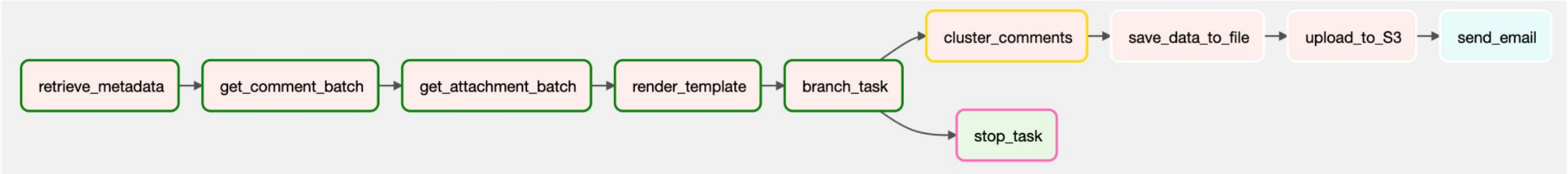


id	title	comment	cluster_kr	cluster_hdbscan	word_count	link
OMB-2023-0001-19683	Comment	I write in support of a new MENA Census category and insist it must include an Armenian subcategory checkbox. Armenian-Americans are among the top three largest MENA communities in terms of population size in the United States and should be represented on the Census.	30	7	44	https://www.reg
OMB-2023-0001-19806	Comment	I write in support of a new MENA Census category and insist it must include an Armenian subcategory checkbox. Armenian-Americans are among the top three largest MENA communities in terms of population size in the United States and should be represented on the Census. <br/>	30	7	45	https://www.reg
OMB-2023-0001-19654	Comment	Armenian-Americans are a sizable and important community that currently is not accounted for in the United States Census and must no longer be excluded. Armenian-Americans are among the top 3 largest MENA communities in terms of population size and must have that reflected with an Armenian checkbox on the upcoming Census form.	10	7	52	https://www.reg

Now includes full text and cluster information

Automating the Workflow

The daily process of downloading, embedding, and clustering comments and making them available to reviewers via a spreadsheet and dashboard was facilitated by the `airflow` package in Python.



Results

Duplicate or near duplicate comments (aka letter-writing campaigns) made up the majority of submissions. There were 82 duplicate comments (with >= 6 exact matches) that made up 68% of all comments. There were only 4,289 comments that weren't an exact duplicate or fall into an HDBSCAN cluster (labeled as 'unique' comments). Grouping comments greatly sped up the work of reviewers.

Letter-writing campaigns for a Middle Eastern and North African (MENA) race/ethnicity category made up 24 of the 43 HDBSCAN identified clusters and those clusters were composed of 13,923 comments.

Comments Received
20,255

"Unique" Comments
4,289

HDBSCAN Clusters
43

Future Directions

Formalize Code – My goal is to generalize the code and package it for others to use. It currently has idiosyncrasies that would require substantial interpretation by users. The plan is for two packages: (1) one to download comments and (2) one for various text analyses, including clustering.

Include Attached Documents – The project did not handle long-form comments in the form of attached PDF or Word documents. Future versions will extract text from documents to be included in analyses. Documents longer than ~300 words present unique challenges for analysis and interpretation.

References

Reimers, N. & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *2019 Conference on Empirical Methods in Natural Language Processing.*

McInnes, L., Healy, J., & Astels, S. (2017) hdbscan: Hierarchical density based clustering. *Journal of Open Source Software*, 2(11).

Contact Info

Brandon Kopp
Senior Data Scientist
Bureau of Labor Statistics
☎ (202) 691-7514
✉ kopp.brandon@bls.gov