Overview

This folder contains data and code necessary for replicating the results of "Estimating the Ideology of Political YouTube Videos."

Computational Requirements

This code was run on NYU's Greene high performance computing (HPC) cluster. It was primarily run on the standard compute nodes while code involving the BERT model used for text-based ideology estimation were run on nodes equipped with GPUs. Further details on HPC can be found here: NYU Greene Supercomputer. The Instructions section contains more information on running the code within this folder.

Data

We collected data on YouTube videos using the YouTube API, Reddit data using the Pushshift API,[Baumgartner et al., 2020] and survey data that was collected for Aslett et al. [2022].

The collected Reddit data is made available in data/included. To ensure compliance with the YouTube API terms of service, we here include video IDs and code for collecting metadata but *not* the actual metadata collected for said videos. For privacy reasons, we do not include the raw survey data and instead include an anonymized data set that only includes a random identifier, the IDs of News & Politics YouTube videos from the anonymous respondent's watch history, and the respondent's political party.

Contents

- **setup.sh**: Once you have installed conda and R, run this script to install the requisite Python and R packages.
- get_packages.R: Installs the requisite R packages.
- data
  - included: Data files provided for replication. These are not overwritten when running the provided replication code. Note that we do not include Youtube video metadata as this requires use of the YouTube API. We do, however, provide video IDs that may be rehydrated by running code included in code/non_replication.
    - human_labeled/*: Contains video pair labels from undergraduate coders.
    - subreddits_by_ideo/*: Hand-labeled political subreddits used as a starting point for detection of more political subreddits.
    - anonymized_video_pid.csv: Contains anonymized respondent IDs, News & Politics YouTube videos from those respondents' watch histories, the predicted ideologies of those videos, and respondent party ID.

    * channels_plot_data.csv: Aggregated channels data for plotting.
    * ideo_channel_means.csv: Means of channels' video ideology scores with channel_ideo indicating the channel label from Hosseinmardi et al. [2020].
    * ng_vids_by_party.csv: Videos from respondents' watch histories grouped by the party ID of the respondents who viewed said videos. Used for Figure 9 in the Supplementary Appendix.
    * new_yt_score_all_subrs.csv.bz2: Raw data for Reddit submissions containing YouTube links.
    * other_bert_preds.csv: Text-based ideology predictions for videos not in our test set and used when validating against human labels.
    * subr_only_el.txt: Subreddit network edgelist.
    * surge_methods_rr.csv: Videos labeled by Surge coders. This data was prepared later on when the paper underwent major revisions.
    * test_set_labels.csv: Video IDs, text-based ideology predictions, and correspondence analysis-based ideology scores. This data was used to evaluate the finetuned BERT model for text-based ideology predictions.
    * training_video_ids.csv: Video IDs and correspondence analysis-based ideology scores for the videos used to train the text-based ideology model.
  - *: The remaining files are intermediate outputs created by running the files in code/ and will be overwritten. Each file is described in the code that produces it.

- code: Main code for reproducing the paper's results with approximate execution time noted in parentheses. To run the Jupyter notebooks, I requested 64 GB of memory and 4 cores.
  - **master_main.sh**: The master script that executes the following files. The step running data_prep.ipynb is initially commented out for quicker testing and replication.
  - data_prep.ipynb ($\sim$ 36 min.): This processes much of the data. Filter the Pushshift data (Reddit posts linking to YouTube videos) for posts with a score > 1 and > 1 comment to aid in the creation of a subreddit network. Create a subreddit-video network and then obtain a subreddit-subreddit network from it. Run community detection on the subreddit network to get a set of political subreddits. Filter the subreddits we found via community detection based on their correspondence analysis components. If they're too far away from the seed set of political subreddits, we exclude them. Finally, create a subreddit-video matrix using the Pushshift data but this time filtered for posts with score ≥ 1 and use logged summed post scores as the entries.
  - get_ca_scores.R ($\sim$ 7 min.): Run correspondence analysis (CA) on subreddit-video matrices with count and score data. We ultimately use the score data but

compute it for count for comparison purposes.

  - human_lab_text.ipynb ($\sim$ 1 second): Calculate the agreement with undergraduate human coders for each video pair for the BERT model ideology estimates.
  - surge_bins.ipynb ($\sim$ 3 seconds): Aggregate and process video labels by Surge coders.
  - master_figures.R ($\sim$ 3 seconds): Produces Figures 2-7 of the main text (Figure 1 is an illustrative diagram generated in LaTeX).

- figures/*: Figures from main text. These will be overwritten by running code/master_figures.R. **Note:** The data used in Figure 4(a) is obtained by running human_lab_text.ipynb and can be found in the output produced by the master script.
- si: This folder contains the data, code, and figures for replicating the results in the Supplemental Appendix.
  - data/*: Contains intermediate data for producing figures.
  - figures/*: Figures from the Supplemental Appendix. **Note:** The data used in Figure 2(a) is obtained by running human_lab_ca.ipynb and can be found in the output produced by the master script.
    * **master_si.sh**: The master script that executes the following files. The step running get_boot_data.R is initially commented out for quicker testing and replication.
    * subsamp_subr_vid_mat.ipynb ($\sim$ 8 minutes):
    * get_boot_data.R ($\sim$ 3 hours using 4 CPU cores): Bootstrap Reddit posts and recalculate the CA ideology estimates 250 times.
    * human_lab_ca.ipynb ($\sim$ 2 seconds): Calculate the agreement with undergraduate human coders for each video pair for the CA ideology estimates.
    * si_figures.R ($\sim$ 9 minutes): Produce Figures 2b, 3, 5, 6, 7, and 9 from the Supplemental Appendix. Figure 1 is produced by code/data_prep.ipynb, Figure 2a is a table containing output from human_lab_ca.ipynb, Figure 4 was produced using video transcripts and thus the corresponding code is in non_replication/topic_models, and Figure 8 was manually produced in Gephi.
- non_replication: This folder contains code used to process data that we do not share publicly and is only for documentation purposes.
  - bert_model
    * train_model.py ($\sim$ 17 hours): Finetune BERT model and take in specified hyperparameters. This code is run via bert.s.

    - get_test_performance.ipynb (∼ 7 minutes): Evaluate model with best validation set performance on the test set.
    - score_vids.py: A utility file for predicting video ideology run by score.s.
    - score_human_lab.ipynb (∼ 1 minute): Use BERT model to predict ideology for videos in our human-labeled set which are not in our subreddit-video matrix.
- data_collection: The runtime of these files will depend on your data collection needs.
    - get_yt_reddit_videos.py: Collect Reddit posts linking to YouTube using the Pushshift API.
    - get_video_ids.py: Parse video IDs from links.
    - get_video_meta.py: Collect metadata associated with collected YouTube video IDs.
- topic_model
    - jb_topic_models_prepped.R: Prepare video metadata transcripts for Structural Topic Models (STM) [Roberts et al., 2019].
    - jb_topic_models_STM.R (∼ 40 minutes): Run STM on video transcripts.
    - transcripts_topic_models_forreplication.R: Plot the topic model results.
- dist_channels.ipynb (∼ 21 seconds): Process channel data and get ideology estimates for figures featuring YouTube channels.
- vid_engagement.ipynb (∼ 2 seconds): Calculate and output engagement stats for videos.
- merge_metadata.ipynb (∼ 9 minutes): Merge video text metadata with videos and the associated CA-based ideology estimates.

Instructions

The code is written in Python, version 3.8.5, and R, version 4.2.0. For Python, the package requirements are contained in `reqs/ytr.yml`. We used version of conda 23.1.0 to manage the Python packages. The BERT models were trained on NVIDIA GPUs.

For R, in addition to the packages in the standard library, we installed FactoMineR version 2.7. You may install the requisite Python and R packages by running setup.sh once you have installed R and conda.

The Python code is primarily in Jupyter notebooks. I used Jupyter Lab with 4 cores and 64 GB of memory and ran the R code in RStudio with 1 core and 16 GB of memory.

To replicate our results using the intermediate data files provided in this folder, run **code/master_main.sh** to replicate the results in the main text of the paper. Run **si/master_si.sh** to replicate the results in the Supplemental Appendix. The Jupyter notebooks may be interactively run with "Run All" in Jupyter Lab and the R code can be run from RStudio after setting the working directory to the folder the code is contained in.

Using the .yml file to recreate the conda environment will produce some warning messages that you can ignore. Do note, however, that if you wish to fine-tune your own BERT model, you will need to install cuda packages that are compatible with your OS/GPUs.

References

K. Aslett, A. M. Guess, R. Bonneau, J. Nagler, and J. A. Tucker. News credibility labels have limited average effects on news diet quality and fail to reduce misperceptions. *Science advances*, 8(18):eabl3844, 2022.

J. Baumgartner, S. Zannettou, B. Keegan, M. Squire, and J. Blackburn. The pushshift reddit dataset. In *Proceedings of the international AAAI conference on web and social media*, volume 14, pages 830–839, 2020.

H. Hosseinmardi, A. Ghasemian, A. Clauset, D. M. Rothschild, M. Mobius, and D. J. Watts. Evaluating the scale, growth, and origins of right-wing echo chambers on youtube. *arXiv preprint arXiv:2011.12843*, 2020.

M. E. Roberts, B. M. Stewart, and D. Tingley. Stm: An r package for structural topic models. *Journal of Statistical Software*, 91:1–40, 2019.